Utilisation de modèles et techniques biostatistiques pour la modélisation des phénomènes de transport

Auteur : Cuenca, Pierre
Promoteur(s) : Cools, Mario
Faculté : Faculté des Sciences appliquées
Diplôme : Master en ingénieur civil architecte, à finalité approfondie
Année académique : 2016-2017
URI/URL : http://hdl.handle.net/2268.2/3281

Université de Liège
Faculté de sciences appliquées,
Master ingénieur civil architecte

Département ARGENCO,
Quartier Polytech 1,
Allée de la Découverte, 9
4000 Liège, Belgique

Utilisation de modèles et techniques biostatistiques pour la modélisation des phénomènes de transport.

Mémoire de fin d'études
par

Pierre CUENCA

Liège
Année académique 2016-2017

Promoteur
Prof Dr. Mario COOLS

Jury
Prof Jacques TELLER, Prof Shady ATTIA, Ismaïl SAADI

Résumé

Les modèles de transport basés sur les activités font intervenir des données organisées sous la forme de séquences. En particulier, les séquences d'activités et les agendas qui peuvent se représenter sous la forme de séquences de lettres. Parallèlement, en biologie, l'étude de séquences et notamment l'étude des séquences d'ADN est un sujet central pour lequel des outils ont été développés dans le cadre de la biostatistique.
Ainsi, fondé sur l'apparente similitude des concepts, l'objet de ce travail est d'étudier la possibilité d'utiliser des outils et des techniques développées dans le domaine de la biostatistique pour la modélisation de phénomènes de transport.
Après avoir parcouru divers domaines en lien avec l'étude du génome, il a été possible d'établir des pistes de réflexion sur des approches visant à en utiliser certaines techniques et méthodologies pour diverses applications dans la modélisation d'agendas et de populations synthétiques dans les cadre des travaux de la planification et de la gestion de la demande de transport.
Un modèle s'inspirant des mécanismes biologiques a été développé afin de démontrer comment en raisonnant par analogie, il est possible d'établir des liens et des passerelles entre les domaines de la biologie et de la modélisation de phénomènes de transport.

Abstract

Activity-based transport models involve data organized as sequences. In particular, activity sequences and agendas which can be represented as a succession of letters. In biology, the study of sequences and in particular the study of DNA sequences is a core subject for which tools have been developed within the framework of biostatistics.
Thus, the aim of this work is to study the possibility of using tools and techniques developed in the field of biostatistics for the modeling of transport phenomena.
After having explored various fields related to the study of the genome, it was possible to establish some possible lines of approach for using some techniques and methodologies for various applications in the modeling of agendas and synthetic population in the framework of transport demand management.
A model based on biological mechanisms has been created to show how, by reasoning by analogy, it is possible to establish links and bridges between the fields of biology and the modeling of transport phenomena.

Remerciements

Je tiens tout particulièrement à remercier mon promoteur de mémoire Mario Cools, professeur associé à l'Université de Liège, qui a assuré l'encadrement de mon travail de fin d'études. En particulier, je le remercie pour sa gentillesse, sa disponibilité et ses encouragements qui m'ont grandement aidé à avancer dans mes travaux de recherche et à approfondir mes connaissances scientifiques.

Je remercie également Jacques Teller et Shady Attia pour leurs avis et conseils qui m'ont permis de m'orienter, de trouver une direction au début de ce travail de fin d'études.

Je veux aussi remercier Messieurs Pierre Leclercq, Jean-Claude Souche, Michel Ferlu et James Olivier pour leur attention, leur soutien et leur bienveillance tout au long de mon parcours dans cette formation d'ingénieur civil architecte.

Je veux également exprimer tous mes remerciements à l'ensemble des équipes enseignantes de l'école des Mines d'Alès et de l'Université de Liège pour la qualité des enseignements reçus et les connaissances qu'ils m'ont transmises.

Enfin, je remercie chaleureusement ma famille et mes amis pour le soutien et le support qu'ils m'ont apporté.

Table des matières

Introduction

Depuis une vingtaine d'années, les modèles basés sur l'activité servant à prévoir la demande de transport n'ont cessé de se développer. Ces modèles reposent sur le constat que les déplacements sont le résultat de la participation à des activités dont les lieux de réalisation sont dispersés dans l'espace. Ainsi, chaque trajet est motivé par la nécessité de participer à des activités qui reflètent les besoins et les désirs des individus et des ménages.

Une caractéristique fondamentale des modèles basés sur les activités est qu'ils étudient les phénomènes de transport en travaillant à l'échelle de l'individu. Dans ces modèles, chaque individu est décrit par un ensemble d'attributs caractéristiques qui définissent un profil particulier. La problématique centrale des modèles basés sur les activités est de définir un agenda "quotidien" pour chacun des individus du modèle sur la base des données de profil. Pour chaque individu il s'agit de prédire les séquences d'activités et les trajets associés ainsi que où, quand, pour combien de temps, comment (quelle chaîne de transport), et avec qui ou pour qui le trajet est effectué, en y associant également les contraintes de temps, d'argent, etc.

Ainsi, la conception de modèles basés sur les activités passe par l'élaboration et le développement d'outils, de techniques et de méthodes afin d’analyser les séquences d'activités et établir les relations entre les différentes entités du modèle : la population d'étude, les individus, les séquences de données traduisant les différents profils, les séquences d'activités, les agendas, etc.

Parallèlement, dans le domaine de la biologie, depuis les vingt dernières années les scientifiques ont perfectionné les méthodes de séquençage de l'ADN et ont élaboré de nombreux modèles statistiques pour décrire les diverses séquences biologiques (ADN, ARN, protéines,...) et établir les relations entre elles ainsi qu'avec les traits phénotypiques des individus. En outre, les recherches menées en biologie dans le cadre de l'étude du génome sont très diversifiées ; progressivement les travaux des biologistes ont permis de décrire plus finement les relations entre les entités du vivant, de procéder à des recherches systématiques de gènes dans les séquences de nucléotides, de prédire les protéines codées par un gène ainsi que leurs fonctions au sein de la cellule, de tracer l'évolution des génomes et des organismes vivants, etc.

Des similitudes frappantes existent entre les domaines de la biologie et du transport :

- les modèles basés sur les activités sont fondés sur les séquences d'activités dont la représentation la plus courante est une écriture sous la forme d'une suite de lettres.
- les travaux des généticiens ont pour élément central la séquence d'ADN, dont la représentation la plus courante prend également la forme d'une suite de lettres (écrite dans un alphabet de quatre lettres).

Faisant ce constat, nous pouvons d’ores et déjà entrevoir des opportunités et envisager la possibilité que parmi les techniques et les méthodes développées par les biologistes, il en serait certaines qui soient transposables à l'analyse et à la création de séquences d'activités de la demande de transport.

De plus, le rapprochement entre séquences d'activités et séquences d'ADN n'est pas le seul que nous pouvons faire. En effet, les deux domaines étudient des populations d'individus associés à des "caractères" propres. Notons encore que dans ces deux domaines d'autres types de séquences font aussi l'objet de travaux de recherches et d'analyses entre lesquels des liens pourraient aussi éventuellement être établis.

Les techniques et les modèles développés en biologie et notamment en biostatistique représentent potentiellement une ressource très intéressante pour la description des phénomènes de transport. La problématique consiste alors à déterminer quel est l'exploitabilité des outils, méthodes et techniques biostatistiques pour la modélisation de phénomènes de transport. Quels sont ceux qui pourraient concrètement être utilisés dans le domaine du transport et pour quels résultats ? En particulier, les outils de travail sur les séquences d'ADN et de protéines peuvent-ils être transposés à la modélisation des séquences de chaînes d'activités ?

L’objet de ce mémoire est d’identifier, dans le domaine de la biologie, des travaux de recherche pour lesquels une transposition des outils ou des méthodes est envisageable. Dans une vision transversale, divers travaux s’avérant présenter un intérêt seront décrits et nous tenterons de mettre en place des éléments de réflexion pour expliquer par quel(s) raisonnement(s) par analogie il serait possible de procéder.
Dans le cadre de ce travail de fin d'études, l'objectif est de fournir des pistes de recherche et de réflexion pour des travaux futurs et de montrer la faisabilité et le potentiel de la transposition de méthodes statistiques du domaine de la biologie vers le domaine de la prévision de la demande de transport.

Dans un premier chapitre, nous ferons un état de l'art du domaine de modélisation de la demande de transport pour présenter les différents travaux, les concepts et les enjeux inhérents à la planification, à la prédiction et à la gestion de la demande de transport. Dans un second chapitre, nous présenterons un ensemble de travaux menés dans le domaine de la biologie pour lesquels des applications peuvent servir à développer ou à améliorer des modèles basés sur les activités. Enfin dans un troisième chapitre, afin de démontrer la faisabilité de la démarche, nous élaborerons un modèle inspiré de mécanismes issus de la biologie en s’appuyant sur un raisonnement par analogie.

Chapitre 1

ETAT DE L'ART : MODELISATION DES PHENOMENES DE TRANSPORT

1.1. Du 4-steps model aux modèles basés sur les activités

1.1.1. Enjeux de la modélisation des phénomènes de transport

Les infrastructures de transport occupent une place prépondérante au sein de l'espace public. L'efficacité de ces infrastructures qui permettent les déplacements de biens et de personnes est un enjeu crucial depuis le début des civilisations. La viabilité et le succès économique des sociétés dépendaient de cette efficacité. C'est encore le cas aujourd'hui avec des enjeux qui se sont diversifiés et complexifiés ; enjeux qui sont à la fois économiques, sociaux, politiques, écologiques, etc.

Il est donc nécessaire d'une part d'être capable d'estimer correctement la capacité des infrastructures (dimensionnement des routes, ponts, tunnels, etc.) et d'autres part, il est nécessaire de pouvoir évaluer les impacts sociaux, économiques et écologiques lors des prises de décisions en terme de politiques de transport et d'aménagement du territoire (création ou modification d'infrastructures, modification du tissu urbain, extension de la ville,...).

Ainsi, les urbanistes et les ingénieurs doivent être en mesure de prévoir les modifications de la demande de transport lorsque les systèmes de transport ou les utilisateurs de systèmes de transport changent.
Pour ce faire, on utilise des modèles, représentations simplifiées de la réalité, qui permettent de déterminer la demande de transport, c'est à dire le nombre de personnes ou de véhicules qui vont utiliser une ou des infrastructures déterminées dans un scenario donné.

1.1.2. Le modèle "classique" à quatre étapes

Le modèle le plus couramment utilisé actuellement est le modèle à quatre étapes ou 4-steps model. C'est un modèle basé sur le "voyage" (ou déplacement) qui constitue l'unité d'analyse. Comme son nom l'indique, ce modèle fait apparaitre quatre étapes successives.

Génération
La première étape est l'étape dite de génération des voyages. A cette étape, l'objectif est d'estimer le nombre de personnes qui vont se déplacer depuis, et également vers chaque zone de l'aire d'étude. L'aire d'étude est divisée en secteurs qu'on associe à leur centroïde respectif. On se trouve dans un modèle où les données spatiales sont agrégées.
Les variables utilisées pour expliquer le nombre de voyageurs immigrant et émigrant de chaque zone sont des attributs socio-économiques tels que la structure familiale, les revenus, le nombre de voitures du ménage, les densités résidentielles, le nombre d'emplois,... Aucune considération d'accessibilité ou relative aux trajets n'est prise en compte.

Distribution - Matrice OD

La seconde étape est la distribution des voyages. A cette étape on détermine les flux ou nombre de voyages entre chaque secteur. Le résultat est l'obtention d'une matrice Origine-Destination (matrice OD) qui indique le nombre de voyages depuis la zone i vers la zone j pour chaque couple (i, j). Il existe deux méthodologies principales pour établir la matrice Origine-Destination : le modèle de facteur de croissance qui extrapole les valeurs d'une matrice O-D préexistante et le modèle par gravité qui passe par la détermination de "coûts" généralement présentés sous forme d'une matrice de coûts. Un coût est déterminé pour chaque couple de zones. Le modèle par gravité se base sur l'ensemble des coûts et des temps nécessaires pour effectuer un voyage et traduit une plus ou moins grande contre-incitation (ou dissuasion) à effectuer une voyage.

Choix Modal

La troisième étape, le choix modal, répartit les voyages selon le type de transport utilisé parmi les choix disponibles (voiture, vélo, train, marche à pied, métro, bus ...). Cette étape est réalisée soit, à nouveau, à l'aide d'un modèle par gravité, soit en utilisant un modèle de choix discret.

Affectation

La quatrième et dernière étape est l'affection des flux ainsi déterminés au réseau associé à chaque type modal. Ainsi, on obtient une prédiction du volume de voyages sur les différentes sections du réseau existant ou envisagé.

Notons que le modèle à quatre étapes est un modèle statique qui reproduit un régime permanent des flux de transport (agrégation temporelle). L'heure du "voyage" n'est pas prise en compte dans le modèle. Généralement, le temps est introduit en appliquant un facteur horaire au volume journalier. Ce facteur est introduit soit à la fin de l'étape de génération, soit à la fin de l'étape d'affectation.

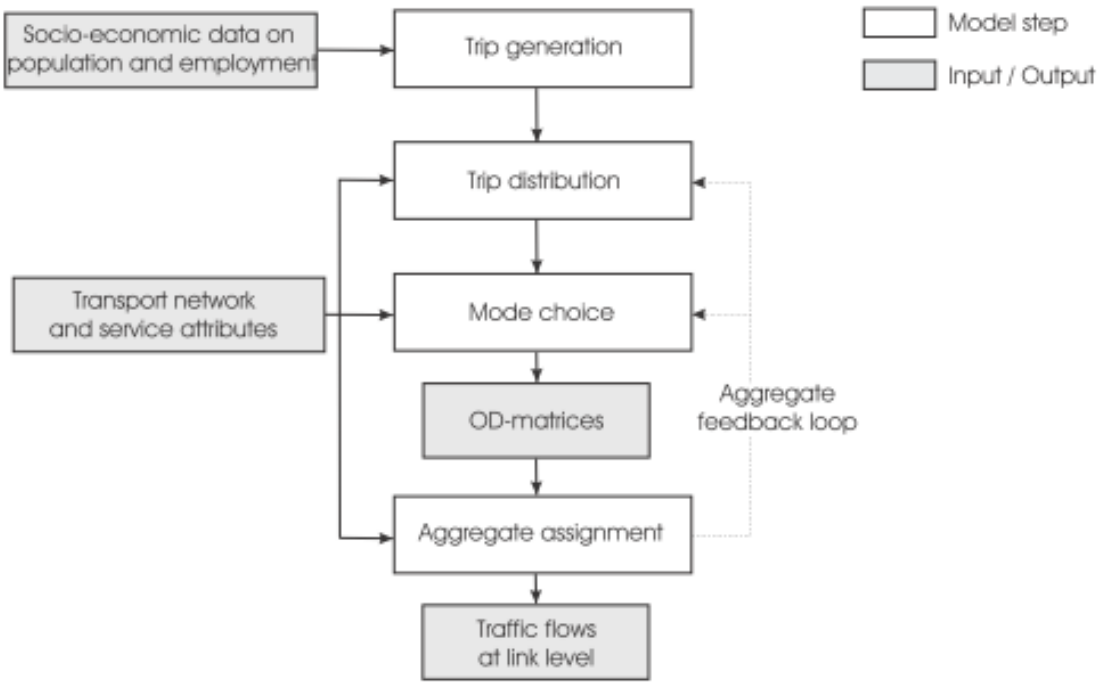


Figure 1 Le modèle à quatre étapes (Source : Feil, 2010 p.7)

1.1.3. Limites du modèle à quatre étapes

Le modèle à quatre étapes offre des résultats satisfaisants lorsque le but est de fournir sur le long terme une offre qui répond à la demande en mobilité. Il permet, en effet, de fournir des résultats acceptables et suffisamment précis lorsqu'il s'agit de dimensionner un élément de réseau de transport ou de déterminer assez grossièrement les effets de premier ordre d'une politique envisagée.

Ainsi, longtemps l'enjeu lors de la planification des transports était d'ordre quantitatif plutôt que qualitatif. Au cours des dernières décennies, de nouveaux paramètres sont apparus tels que l'augmentation des coûts de construction de nouvelles infrastructures, la sensibilité quant aux problèmes de congestion et de pollution. D'une logique de satisfaction de la demande, on est passé à une logique de gestion de la demande. Le but est désormais de comprendre les besoins des usagers en déterminant pourquoi, quand, où et comment les personnes se déplacent afin de proposer des moyens de transport adaptés mais également afin d'influencer les comportements des usagers notamment en promouvant les transports en commun et la mobilité douce.

Or le modèle à quatre étapes présente de nombreuses limites et n'est plus un outil efficace pour déterminer les nouvelles politiques menées en planification des transports.
Quatre principaux écueils ont plaidé pour l'abandon de ce modèle au profit du développement de modèles basés sur l'activité (d'après Rasouli et Timmermans, 2014).

Le premier problème relevé est le manque d'intégrité. Le modèle étant composé de sous-modèles indépendants entre eux, il serait souhaitable qu'il y ait une congruence et une cohérence entre les différents sous-modèles. Or ce n'est pas toujours le cas. En effet, les temps de trajet qui proviennent de l'étape d'affectation au réseau, ne sont pas systématiquement cohérents avec les temps de trajet utilisés comme données d'entrée pour prédire la destination des voyages à la deuxième étape. Et ceci n'est qu'un exemple parmi d'autres.

Un autre reproche fait au modèle à quatre étapes est qu'il est fortement agrégé par nature. Les données spatiales et temporelles sont agrégées ainsi que les voyageurs. Cela permet de réduire la complexité computationnelle mais en contrepartie, cela empêche d'avoir une finesse dans l'interprétation des résultats. Les points de départ et d'arrivée des voyages sont ramenés à un seul point de l'espace pour chaque zone. Le modèle permet seulement de distinguer les heures de pointe et les heures creuses en supposant que le flux de voyageurs est constant pour chacun de ces deux cas de figure (régime permanent). Enfin, toutes les personnes ou tous les ménages dans une zone sont considérés comme identiques étant ramenés à un individu ou respectivement à un ménage moyen. Toutes ces agrégations créent des biais d'agrégation dus au fait que les interactions entre individus sont minimisées voir ignorées dans ce modèle.

Non seulement, le modèle ne prend pas en compte les interactions entre ménages (ou individus) mais il fait une hypothèse d'indépendance spatiale et temporelle entre les déplacements effectués par une même personne.
Le modèle ne tient compte d'aucune dépendance entre les déplacements qui appartiennent à une même chaîne (exemple : maison-travail-maison) ; les contraintes de temps ou la cohérence du mode de transport ne sont pas prises en compte. Le modèle n'intègre pas non plus les interactions entre les membres d'un même foyer ; un membre du foyer peut effectuer une action pour l'ensemble du foyer (comme par exemple les courses alimentaires).

Enfin, le modèle manque de réalisme dans la mesure où il ne tient pas compte des comportements humains. Dans sa conception même, le modèle est construit afin de déterminer l'intensité des flux de voyageurs. Dans cette optique, le problème est abordé d'un

point de vue de physiciens qui appliquent, par analogie, les lois de la thermodynamique. Le modèle à quatre étapes est analogue à un modèle de physique ; il ne se base sur aucune considération sociale et ne fait apparaitre aucun mécanisme comportemental de prise de décision.

1.1.4. Une nouvelle façon d'aborder le problème : les modèles basés sur l'activité

Comme évoqué précédemment, la planification du transport est passée d'une logique de dimensionnement et de satisfaction de la demande à une logique de gestion de la demande. Afin de mener de telles politiques il est nécessaire de bien cerner les besoins de mobilité de chaque usager. De plus, le but n'est plus seulement de répondre à la demande mais aussi d'influencer les choix des personnes qui se déplacent. Dans ce but, il est nécessaire de s'intéresser aux comportements individuels des usagers et aux mécanismes de prise de décision.

Les modèles basés sur l'activité se sont progressivement imposés comme alternative au modèle à quatre étapes.

Ces modèles sont "basés sur la compréhension du fait qu'un déplacement provient de la nécessité de participer à des activités, qui à leur tour traduisent des besoins, des désirs et des engagements des individus et des ménages. Le but fondamental des modèles basés sur l'activité est de prédire quelle séquence d'activités et quels déplacements associés sont réalisés par (tout) individu d'un foyer, où, quand, pour quelle durée, la chaine de mode de transport impliquée, et possiblement avec et pour qui, soumis à un ensemble de contraintes spatiales, temporelles, institutionnelles, spatio-temporelles et possiblement de budget." (D’après Rasouli et Timmermans, 2014)

Les modèles basés sur l'activité prennent l'individu comme unité de travail et sont de fait, généralement des modèles désagrégés. Ils sont de plus fondés sur les besoins humains et les mécanismes de prise de décision, et tiennent donc compte du comportement humain. Ainsi, à priori, ces modèles permettent de dépasser les limites relevées dans le modèle précédent.

1.2. La création d'agenda : une problématique centrale

L'enjeu majeur pour créer des modèles basés sur l'activité sera de générer pour chaque acteur un plan d'activité complet, c’est à dire un agenda.

Dans le cadre de la modélisation des phénomènes de transport, un agenda est une chaîne d'activités ordonnées, qui définit en plus pour chaque activité une localisation, une durée et une heure de début (et/ou de fin). Certaines informations complémentaires peuvent être ajoutées telles que par exemple les modes de transport utilisés pour effectuer les différents trajets ou les individus participants à l'activité conjointement (notamment les autres membres du foyer).

Le problème fondamental réside dans la grande pluralité d'agendas formulables pour un seul et même individu. En observant le monde réel, on se rend compte qu’il existe autant d’emploi du temps différents qu’il y a d’individus (activités réalisées, séquence de réalisation, heures et lieux). Or, chacun fait des choix parmi une quantité de possibilités qui s'offrent à lui en omettant une grande partie des solutions qu'il ignore ou néglige.

Certes, le but n'est pas de donner une vision extrêmement précise de la réalité mais de proposer un modèle. Néanmoins, en ne prenant en compte qu'une liste de dix activités, il existe près de dix millions de séquences d'activités possibles ($10! + \sum_{i=1}(10 - i)!\ C_{10}^{i} = 9{,}86E^{6}$). A cela s'ajoute encore le problème de localisation et de programmation dans le temps, ce qui étend encore le champ des possibles qui croit exponentiellement (cf. formule de Stirling). Ainsi, le problème ne pourra en aucun cas être résolu en passant par une énumération exhaustive des solutions.

Dimension	Number of alternatives	Number of alternatives per activity agenda
Activities (e.g., per day)	10	10
Sequencing alternatives		10!
Timing alternatives	100 per activity	1000
Location alternatives	100 per activity	1000
Mode choice alternatives (without tour constraints)	5 per activity	50
Route choice alternatives	10 per activity	100
Total		**10^{17}**

Figure 2 Un exemple pour illustrer la dimension du problème combinatoire (source : Feil, 2010, p.14)

Ce n'est pas pour autant un problème insolvable. Après tout chacun de nous chaque jour est capable de planifier ses activités avec un résultat relativement satisfaisant. Dès lors toute la difficulté et tout l'enjeu est de résoudre ce problème automatiquement par un procédé informatique, pour tout individu et en restant dans des temps de calcul computationnel acceptables.

Les travaux des sociologues F.S. Chapin et T. Hägerstrand sur l'étude des séquences d'activités humaines ont servi de point de départ pour la création d'agendas.

1.3. Les travaux de recherche s'intéressant au comportement

Bien que depuis de nombreuses décennies, la demande de voyage ait suscité de très nombreuses études, la modélisation des systèmes de transport basée sur l'enchaînement des activités est récente. Fondée sur des travaux de recherches initialement liés aux études de l'Espace géographique (évolution démographique et utilisation des sols), le développement théorique de l'approche basée sur l'activité a été initié au début des années 1970 par F. Stuart Chapin et Torsten Hägerstand. A l'origine de ce changement de paradigme, ils ont posé les bases des théories découlant du constat que la demande de voyage d'un individu est motivée par le désir ou le besoin de participer à des activités.

1.3.1. La théorie de F. Stuart Chapin

Dans les années 1960, F.S. Chapin a mené des études sociologiques du comportement humain dans le cadre sur la planification du territoire. Dans ses travaux, il soutient que les aménagements urbains n'ont d'autre intérêt que d'offrir aux individus la possibilité d'exercer leurs activités désirées. Le tissu urbain n'est pas créé en suivant des principes dogmatiques d'aménagement mais comme une conséquence des schémas d'activités. Ce sont les schémas d'activités quotidiennes qui influence les comportements de localisation qui eux-mêmes définissent l'organisation du territoire.

Notons au passage que F. S. Chapin n'a pas abordé la relation entre les activités et les déplacements. Les déplacements ne semblent pas être pour lui un élément central des choix de localisation, ils en sont une conséquence. S'ils sont une composante clé de la planification des transports, ils n'entrent pas explicitement en compte dans la planification des sols.
Ceci étant posé, il sera nécessaire que les outils de planification des sols tiennent compte des activités humaines, ce qui antérieurement à Chapin n'était pas le cas. La structure urbaine était définie selon des théories économiques normatives et de maximisation du profit.

Dès lors, il est essentiel de comprendre les raisons qui mènent l'individu à réaliser des activités. Selon F.S. Chapin, le comportement d'activité est fortement influencé par les motivations inhérentes à la survie, la sécurité, la réussite, le statut social et aux besoins relatifs au bien-être, soit finalement l'ensemble des besoins répertoriés dans la pyramide de Maslow. Sa théorie repose sur le principe que les individus choisissent leurs activités parmi un ensemble d'opportunités. Ces choix sont formulés en tenant compte de leurs motivations.

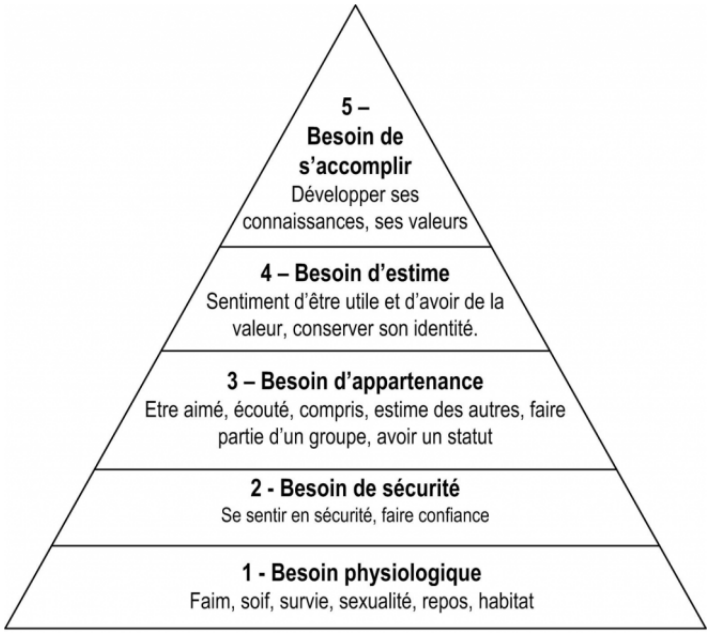


Figure 3 Pyramide de Maslow
(source :http://www.bing.com/images/search?q=&view=detailv2&id=122592870801B0F0877E5FEB844CC08E05B5DF37&ccid=lkMYjerW&iss=fav&cbn=favorites&selectedIndex=0&FORM=SVIM01)

Pour F.S. Chapin toute action ou activité est motivée par un objectif. Toute motivation est considérée comme positive. Par exemple, l'action d'aller travailler n'est pas perçue comme une obligation ou une contrainte, elle est motivée par le besoin d'appartenance à un groupe, le besoin de reconnaissance et d'estime de soi et bien sûr la rémunération.

L'élaboration de la théorie émise par F.S. Chapin repose principalement sur l'expression du comportement lié à la motivation de l'individu. Certains comportements sont d'origine purement physiologique mais d'autres sont de nature plus complexe. Ces derniers résultent de choix individuels influencés par certaines caractéristiques personnelles. F.S. Chapin met ainsi en avant la propension, l'opportunité, la situation et le contexte environnemental comme étant les principales sources d'influence entrant dans le processus de choix individuels.

La propension est le penchant qu'aura un individu à participer à une activité. Cette "tendance naturelle" est le reflet de la prédisposition l'individu à participer à un type d'activité et de ses aspirations émanant de l'expérience vécue.

Les opportunités sont définies par des facteurs physiques et spatiaux offrant aux individus des occasions favorables pour s'engager dans des activités particulières.

La situation permet d'évaluer si les conditions et les circonstances du moment sont favorables à la participation à une ou plusieurs des activités disponibles, en prenant notamment en compte leur agencement dans la séquence d'activités quotidiennes.

Le contexte environnemental se réfère aux facteurs contextuels ayant une influence sur les décisions et les attitudes des individus. Les incitations économiques, les considérations vis-à-vis de l'environnement et des politiques publiques peuvent avoir des répercussions sur le comportement du choix.

Ainsi, F. Stuart Chapin a suggéré une approche fondée sur la motivation de l'individu et a ainsi avancé que les modèles de participation aux activités révélées découlent de l'interaction des contraintes sociétales et des motivations individuelles inhérentes.

1.3.2. Torsten Hägerstrand et le concept des Contraintes

Dans le même temps, les travaux de Torsten Hägerstrand, professeur suédois en géographie sociale et économique, connu pour son travail sur la migration, la diffusion culturelle et la géographie du temps, tendent à conceptualiser la relation entre la participation à l'activité et les concepts espace-temps.

Ainsi T. Hägerstrand (1970) émets pour théorie que les modèles d'activité des individus dans le contexte multidimensionnel de l'espace et du temps, sont inhérents à la prise en compte simultanée de trois types de contraintes différentes que sont les contraintes d'autorité, de capacité et de couplage. Il souligne en effet que les individus sont restreints dans leurs actions. Ils ne peuvent pas effectuer toutes les actions qu'ils souhaitent parce qu'ils sont limités par ces trois types de contraintes.
Les trois contraintes que T. Hägerstrand met en évidence sont :

1. Capability constraints

Les contraintes de capacité constituent des limites qui proviennent d'une certaine "incapacité" de l'individu à réaliser une action. Elles limitent les activités de l'individu en raison de causes naturelles comme par exemple le temps qu'il doit consacrer à satisfaire des besoins physiologiques tels que par exemple manger, boire, se reposer ou dormir, mais aussi en raison de ses capacités et aptitudes physiques personnelles (personnes âgées par exemple). Ces restrictions peuvent également découler d'un manque de moyens matériels ou d'outils appropriés nécessaires à la réalisation de l'activité (par exemple la non-possession d'une bicyclette et/ou d'un véhicule à moteur, matériel de sport spécifique, moyens financiers, etc.).

Selon Torsten Hägerstrand (1970), dans son approche de la géographie du temps, les restrictions émanant des contraintes de capacité permettent d'expliquer l'activité spatiale des individus et ainsi d'en élaborer des prismes espace-temps représentatifs. Considérant par exemple le moyen de locomotion, dans un laps de temps donné, une personne se déplaçant en voiture aura un "rayon d'action" nettement plus large qu'une personne n'ayant d'autre moyen que de se déplacer à pied.

Les personnes pouvant se rendre à une activité en utilisant le moyen de transport le plus rapide, soit le plus efficace, auront des prismes espace-temps plus élargis que celles pour lesquelles il faudrait plus de temps pour parcourir une distance moindre (par exemple l'utilisation d'une voiture personnelle versus la marche à pied).

2. Coupling constraints

Les contraintes dites de couplage sont relatives aux diverses restrictions pouvant émaner de la dépendance d'un individu aux autres. Les contraintes de couplage définissent où, quand et la

durée des activités devant être planifiées dans le but de permettre à une ou plusieurs autres personnes d'y participer.

C'est ainsi par exemple le cas pour la participation à un sport d'équipe, une chorale, un groupe musical ou autres, pour lesquels il sera par définition requis d'être un certain nombre d'individus pour constituer un groupe ou une équipe. C'est encore aussi le cas pour certains sports, même réputés individuels comme le tennis, pour lequel il faut être à minima deux pour pouvoir jouer.

Il peut aussi s'agir d'une dépendance matérielle comme par exemple dans le cas d'une personne ne possédant pas de voiture. Afin de participer à des activités, en fonction de sa situation géographique personnelle et de la localisation des équipements et infrastructures, outre l'usage des transports en commun, il aura éventuellement la possibilité de recourir au co-voiturage. De ce fait, pour palier des contraintes de capacité, l'individu pourra alors faire face à des contraintes de couplage.

Ces divers exemples traduisent la nécessité d'une prise de rendez-vous, le besoin d'une coordination des agendas personnels de manière à permettre aux divers acteurs ainsi qu'aux outils ou matériels, d'être disponibles et rassemblés à un certain moment en un endroit donné pour une durée définie. Quelle qu'en soit la raison ou la cause, la dépendance d'un individu ou l'interdépendance des individus entre eux, induisent nécessairement des contraintes organisationnelles dont le caractère restrictif, pouvant même aboutir sur une impossibilité d'agir, est représentatif des contraintes de couplage telles que les a définies T. Hagerstrand dans son approche de la géographie du temps pour expliquer l'activité spatiale des êtres humains.

3. Authority constraints

Les contraintes d'autorité sont représentatives des restrictions qui s'imposent à l'individu par les contraintes légales, la vie sociale étant soumise au respect de règles institutionnelles s'inscrivant dans le cadre juridique dans lequel il évolue. Ces limitations s'imposent à lui dans ses opportunités spatiales et temporelles de participer à des activités sans qu'il puisse en influencer le cours.

Ainsi par exemple, si vous n'êtes pas détenteur d'un permis de conduire, vous n'êtes pas autorisé à conduire un véhicule, mais cela peut aussi être le cas dans le cadre de mesures environnementales. Par exemple de plus en plus, de nos jours, en zones urbaines, que ce soit de manière ponctuelle, provisoire ou durable, des mesures restrictives, telles que la circulation alternée ou l'interdiction des véhicules diesel en centre-ville, sont mises en place. Ces mesures constituent ainsi des contraintes d'autorités en ce sens qu’elles ne vous autorisent pas à utiliser votre voiture ou alors seulement de manière restrictive, conformément aux règles et critères définis.

Plus simplement encore, un autre type de contraintes d'autorités émanant des lois du travail, d'accords d'entreprise et autres décisions managériales ou politiques, sont les horaires d'ouverture et de fermeture des infrastructures, des magasins, des établissements publics ou privés. En d'autre termes, elles expriment le fait qu'un individu ne pourra pas accéder à l'activité en dehors de la plage horaire autorisée, ce qui constitue une composante restrictive de la décision de participation à l'activité et dans l'élaboration de l'agenda.

Il était primordial de rappeler ici les notions essentielles évoquées par Torsten Hägerstrand (1970) dans sa conceptualisation des modèles d'activité individuels. Le quotidien des individus consiste en la réalisation de projets jalonnés d'actions conditionnées par trois types de contraintes majeures que nous avons tenté d'expliciter ci-dessus. Ce concept définissant la

mécanique temps-espace des contraintes est d'importance capitale dans l'interprétation des différents chemins de vie suivis ou rejetés par chacun des acteurs. Ces contraintes "spatio-temporelles autorisées" créent un prisme tridimensionnel et constituera le fondement théorique des modèles "prismes spatio-temporels" plus tard mis en œuvre à compter des années 1990.

1.3.3. Les Prismes spatio-temporels

Après avoir passé de nombreuses années à étudier les modèles de migration humaine, Torsten Hägerstrand est convaincu de la nécessité d'examiner conjointement les coordonnées spatiales et temporelles de l'activité humaine. Soutenant qu'une compréhension du comportement spatial désagrégé est primordiale, il dévoile en août 1969, son modèle spatio-temporel destiné à changer le cours de l'histoire dans les sciences sociales.

Ainsi T. Hägerstrand a proposé le concept de trajectoire spatio-temporelle pour illustrer la façon dont une personne navigue à travers l'environnement spatio-temporel. Il utilise la notion de chemin dans l'espace-temps pour démontrer comment les activités spatiales des individus sont essentiellement régies par l'existence de contraintes qui se produisent à différents niveaux pour produire des hiérarchies d'accessibilité et que les décisions d'accès aux activités résultent de la corrélation entre les capacités spatiales et temporelles des individus.

Dans ces travaux T. Hägerstrand représente les activités des individus par des chemins de vie dans l'espace-temps. Un chemin de vie est un dispositif graphique qui retrace l'historique de la localisation d'une personne dans l'espace à chaque instant.

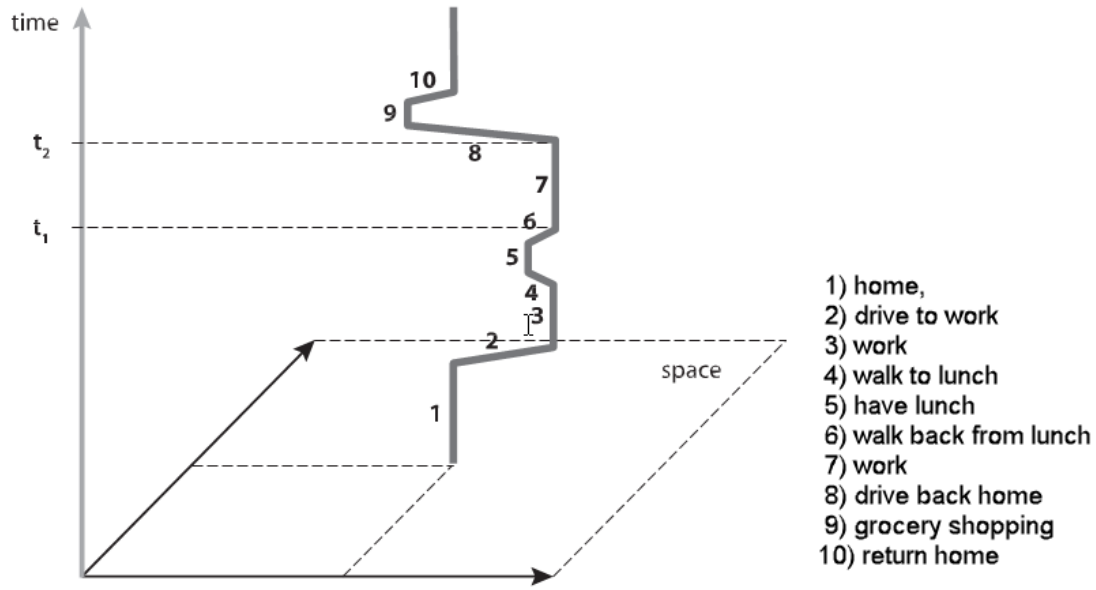


Figure 4 Visualisation d'un chemin de vie dans l'espace-temps – (Martin Šveda, Michala Madajová, 2012)

Les lignes verticales dans les chemins d'activités traduisent le fait que l'individu reste un certain temps dans un même endroit. Ces verticales sont parfois aussi nommées stations. L'inclinaison de la trajectoire de l'individu montre la relation entre son mouvement dans l'espace et le temps nécessaires à ce mouvement. Plus le segment tendra à être horizontal, plus l'individu passe d'une position géographique à une autre rapidement. Considérant les chemins de plusieurs individus d'une population, certains chemins peuvent converger vers une même station, le groupe de chemin ainsi formé constitue un bundle. Les bundle tendent à être interdépendants.

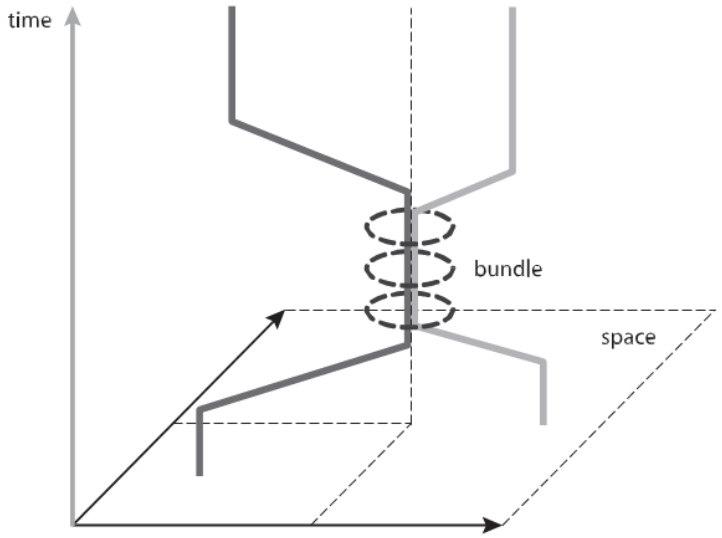


Figure 5 Visualisation d'un bundle – (Martin Šveda, Michala Madajová, 2012)

De même que les chemins de vie sont utilisés pour décrire graphiquement le mouvement des individus, le groupe Lund, formé de Hagerstrand et d'autres professeurs de l'université de Lund, a développé la notion de prismes spatio-temporels. Un prisme spatio-temporel est défini comme la zone qu'un individu peut atteindre dans une certaine fenêtre de temps. Ce concept défini ainsi l'espace atteignable par un individu dans un intervalle de temps donné et contraint donc les choix de destination qu'une personne peut faire.

Les stations sont utilisées comme points de référence. Hagerstand définit le prisme sur une journée comme un ensemble de plusieurs petits prismes, chacun associé aux différentes stations qui jalonnent le chemin de vie quotidien d'un individu. Ainsi, le prisme d'un individu est un amalgame de point de départ, une contrainte de vitesse, les projets et les activités à effectuer et les contraintes de couplage imposées par les limites de temps-espace. La forme du prisme est circonscrite par la vitesse de déplacement de l'individu et si la situation d'origine est la même que la situation de destination ; si c'est le cas, le prisme est symétrique, sinon il sera asymétrique. La projection du prisme sur le plan géographique donne l'aire des trajets potentiels (Lenntorp 1976), qui détermine toutes les positions géographiques atteignable par un humain dans un temps donné.

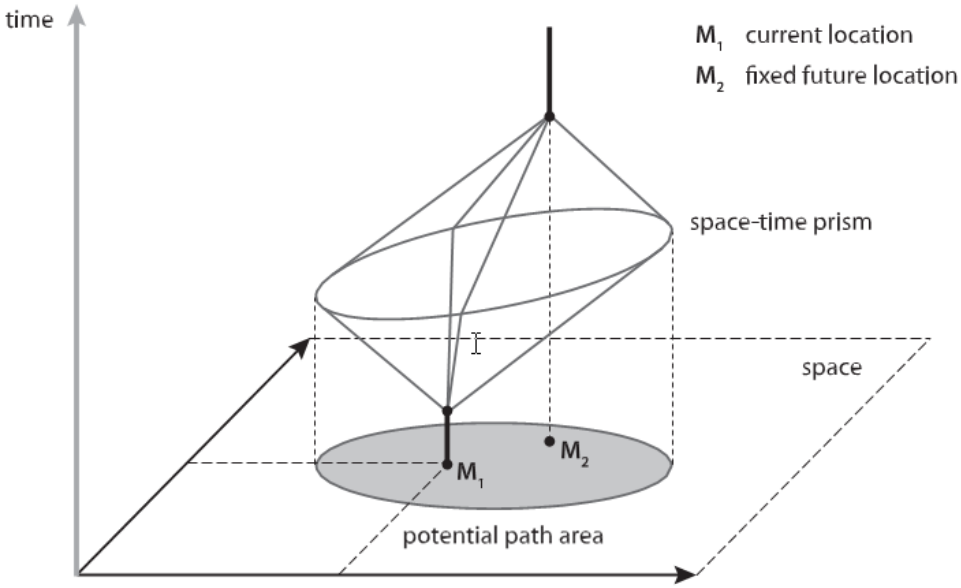


Figure 6 Visualisation d'un prisme – (Martin Šveda, Michala Madajová, 2012)

Les prismes ne sont pas seulement utilisés au niveau individuel, mais aussi pour analyser l'affectation du temps et l'occupation spatiale, classiquement et différemment appelés «utilisation des terres». Ils peuvent être très utiles en tant qu'outils analytiques.

Permettant de mieux appréhender l'activité humaine dans un environnement espace-temps, le concept d'espace-temps est un outil puissant parce qu'il est simple. Cet outil a trouvé de larges applications dans de nombreux domaines dont ceux de la demande de transport et de l'urbanisme. Ces modèles basés sur les contraintes sont utilisés pour leur capacité à identifier des calendriers d'activités réalisables dans un contexte temps-espace changeant.

Plus de quarante ans après, la conceptualisation des modèles spatio-temporels initiée par T. Hägerstrand continue d'influencer les travaux de recherche en sciences sociales.

1.3.4. Evolution des théories

Les approches antagonistes et malgré tout complémentaires proposées par F.Stuart Chapin et par Torsten Hägerstrand, forment la base d'une grande partie de la recherche sur l'analyse de l'activité. Leurs travaux ont été exploités par de nombreux autres sociologues qui ont développé des théories émanant d'une combinaison des deux approches.

En 1975, Cullen et Godson proposent un compromis aux approches antérieures. Dans leur approche, ils introduisent une notion de flexibilité et d'inattendu par rapport au modèle de T. Hägerstrand.
Ils distinguent des activités de routines, non flexibles, qui définissent un cadre structurant à l'organisation de l'agenda quotidien. Entre ces plages d'activités dont les lieux et les horaires sont fixés, il demeure des plages de temps libre.

Une fois les activités de routines planifiées, il reste à définir les activités réalisées pendant les périodes de temps libre. Cullen et Gudson introduisent la notion de priorité et de choix délibéré. Les priorités sont définies sur la base des travaux de F.S. Chapin sur la propension, l'opportunité, la situation et le contexte environnemental ; l'ordre de priorité des activités dépendra notamment des goûts et des aversions réels de l'individu pour chacune des activités. Malgré tout, la séquence d'activités menées restera soumise au processus des contraintes spatiales et temporelles établi par T. Hägerstrand.

En 1983, dans le cadre de travaux de recherche visant à améliorer les modèles de demande de transport, Jones, Dix, Clarke et Heggie combinent à leur tour les approches de Chapin, Hägerstrand et Cullen et Godson en prenant le ménage comme unité d'étude et non plus l'individu isolé. Ils mettent un accent particulier sur les contraintes de couplage dans la réalisation des agendas.
Leur démarche méthodologique est similaire à celle de Cullen et Godson. Les activités relatives au ménage sont planifiées autour des obligations de chacun de ses membres. Dans leur conception, Jones et al. identifient des contraintes liées aux activités (durée, …), des contraintes d'accessibilité (heures d'ouverture, localisation,…), des contraintes liées au ménage (besoins des autres membres du ménage,…) et des contraintes individuelles (engagements,…). A ces contraintes s'ajoutent enfin des contraintes d'ordonnancement inhérentes à l'interdépendance des individus constituant le ménage.

1.4. Méthodes de modélisation

Au cours des trois dernières décennies, les chercheurs ont exploré de nombreuses pistes et ils ont développé une grande variété de modèles cherchant constamment à dépasser les limites des modèles précédents. Il est néanmoins possible d'établir des similitudes entre certains modèles qui suivent des logiques de modélisation analogues ou présentent des caractéristiques communes. On peut ainsi établir une classification des modèles.

Classification des modèles basés sur l'activité

Modèles désagrégés				*Modéles agrégés*	
Modèles basés sur l'utilité		Processus Computationnel	Autres	4-step model amélioré	Equation structurelle
Econométrique	Microsimulation				
- Bhat	- Starchild - PCATS - CEMDAP - TRANSIMS - MATSIMS	- Scheduler - AMOS - Albatross	- HAPP - HARP - AURORA	- VISEM	- Golob

Figure 7 Classification des modèles basés sur l'activité (D'après Feil,2010)

1.4.1. Les modèles économétriques basés sur l'utilité

Les modèles économétriques sont des modèles qui suivent la théorie basée sur le postulat que les individus maximisent l' "utilité" lorsqu'ils choisissent une alternative parmi la multitude d'agendas possibles.

Dans les modèles classiques (les modèles basés sur les "voyages"), une valeur d'utilité est associée aux attributs liés au voyage tel que par exemple le mode de transport. Dans les modèles basés sur l'activité, cette notion d'utilité sert à caractériser la performance des activités, ou d'une séquence d'activités.

Les modèles qui exploitent ce principe d'utilité reposent sur deux étapes. Dans un premier temps est créé un ensemble de choix qui contient un ensemble (plus ou moins important selon les modèles) de solutions réalisables. Puis dans un second temps, la meilleure alternative est choisie parmi l'ensemble de solutions. Le but de cette méthode est de reproduire le processus de décision d'un individu qui choisit la meilleure alternative parmi l'étendue des possibles.

Les modèles économétriques travaillent avec un ensemble de choix très étendus. Dans ces modèles l'ensemble des solutions (faisables) est considéré, ce qui revient à considérer que l'individu prend en compte l'ensemble des alternatives qui s'offrent à lui.

Ensuite, la probabilité d'être choisi comme la meilleure alternative est calculée pour chacun des choix de l'ensemble de solutions. Une ou des équations sont utilisées pour décrire les relations entre attributs (les attributs sont des données d'ordre socio-économique, des données liées aux activités et des données liées au trajet). Ces modèles sont étroitement liés aux modèles de choix discret qu’ils utilisent et enrichissent. Les modèles les plus couramment utilisés sont les modèles logit multinomiaux et les modèles logit à nœuds.
La distribution de probabilité peut également être traduite comme une solution alternative spécifique via une simulation de Monte Carlo.

Les modèles économétriques ont pour avantage de reposer sur des modèles statistiques bien établis et relativement fiables. Mais, ils deviennent rapidement complexes et nécessitent une grande puissance de calcul ce qui oblige à limiter le niveau de détail des modèles.

Quelques modèles et/ou auteurs de référence :

- Bhat, C.R., tous les travaux entre 1997 et 1998
- Bhat, C.R. and S.K. Singh, 1999
- **The daily activity schedule model** (Ben-Akiva & Bowman, 1998; Ben-Akiva, Bowman, & Gopinath, 1996; Bowman, 1995, 1998; Bowman & Ben-Akiva, 2000; Bowman, Bradley, Shiftan, Lawton, & Ben-Akiva, 1998)
- **Tasha** (Miller & Roorda, 2003; Roorda, 2005; Roorda, Doherty, & Miller, 2005; Roorda & Miller, 2005, 2007; Roorda, Miller, & Habib, 2008)

1.4.2. Les modèles basés sur l'utilité reposant sur la microsimulation

Les modèles qui reposent sur la microsimulation sont également des modèles basés sur l'utilité et le principe de maximisation de l'utilité. Ils se distinguent des modèles économétriques par le fait qu'ils utilisent une méthode séquentielle qui fait apparaître des étapes successives de prise de décisions. Les modèles de microsimulation visent à réduire le nombre de solutions qui composent l'ensemble de choix.

A chaque étape décisionnelle le modèle recours à des modèles de choix discret (modèles logit multinomiaux et modèles logit à nœuds) ou à des modèles de hasard qui permettent de choisir une alternative parmi un ensemble de choix restreint.

Par ailleurs, en plus de séquencer le processus de création du meilleur agenda, les modèles de microsimulation utilisent l'ensemble des travaux en lien avec les théories développées par Hägerstrand qui soulignent le rôle prépondérant des diverses contraintes dans l'organisation quotidienne des activités et des déplacements. En mettant l’accent sur les contraintes qui influencent et qui structurent les plannings d'activités quotidiens, les chercheurs réussissent à réduire l'ensemble des possibles et donc l'ensemble de choix dans les modèles de microsimulation.

Quelques modèles et auteurs de référence :

- **STARCHILD** (Recker et al., 1986)
- **PCATS,** Prism-Constrained Activity Travel Simulator (Kitamura & Fujii, 1998) & **FAMOS** (Pendyala, Kitamura, Kikuchi, Yamamoto, & Fujii, 2005)
- **CEMDAP** (Bhat, Guo, Srinivasan, & Sivakumar, 2004)
- **TRANSIMS,** TRansportation ANalysis SIMulation System (Smith, Beckman, Anson, Nagel, & Williams, 1995; Hobeika, 2005)

1.4.3. Les processus computationnels

Dans les deux types de modèles précédemment évoqués, le principe est de chercher un optimal parmi l'ensemble des solutions en faisant émerger la solution qui maximise l'utilité. Afin de mimer le comportement décisionnel humain, toutes les solutions sont prises en compte et c'est parmi cet ensemble qu'est recherchée la meilleure solution qui sera la solution retenue. Or, si on peut en effet considérer que le processus de décision d'un individu consiste à choisir une alternative qui est d'après lui, la meilleure parmi les différentes options qu'il envisage, il semble irréaliste de prétendre qu’un individu prend systématiquement la meilleure décision ni même qu'il puisse déterminer laquelle est la plus optimale. Par ailleurs, il est également irréaliste d'envisager qu'il prenne en considération tout le champ des possibles.

Les méthodes qui mettent en œuvre un processus computationnel visent à pallier à ce manque de réalisme en ne recherchant plus la solution optimale mais, seulement, une solution acceptable, à la fois réalisable et réaliste. Cette solution est construite à travers un processus heuristique de prise de décision suivant le contexte. Le modèle repose en outre sur des principes comportementaux de prise d'informations, de traitement de l'information et d'utilisation de l'information.

Les modèles sont construits sur la base de règles de décisions (souvent de type "if-then") et miment la démarche intellectuelle d'un individu qui essaie une séquence de possibilités et qui choisit la première qui convient.

Quelques modèles et auteurs de référence :

- **CARLA** (Jones et al., 1983)
- **SCHEDULER** (Gärling et al., 1989; Golledge et al., 1994)
- **ALBATROSS** (Arentze et Timmermans, 2000, 2004, 2005, 2011)

1.4.4. Les modèles agrégés

Les modèles économétriques, les modèles de microsimulation et les processus computationnels sont les trois grands types de modèles qui travaillent avec des données désagrégés. Ils prennent l'individu comme unité d'étude et permettent de générer un agenda par individu. Il existe des modèles basés sur l'activité qui utilisent des données agrégées.

Les modèles agrégés ne travaillent pas sur la base de voyageurs individuels mais travaillent avec des groupes de voyageurs en faisant l'hypothèse d'une répartition homogène des profils et des comportements au sein des différents groupes.

Il existe deux types de modèles agrégés : des modèles qui développent le modèle à quatre étapes et des modèles qui reposent sur des équations structurelles.

Modèles à quatre étapes étendus
Les modèles qui reprennent les étapes du modèle classique à quatre étapes introduisent la notion d'activité à la première étape du modèle pour la génération des voyages. Le nombre de voyages est déterminé d'après la probabilité que certaines chaînes d'activités ont d'être observées pour la population de chaque zone. Les voyages sont ensuite distribués en utilisant une fonction de déterrence relativement classique si ce n'est qu'elle est déterminée pour l'activité menée. Ensuite un autre sous-modèle permet de définir le mode de transport pour les "agendas".

Modèles basés sur des équations structurelles
Le second type de modèle est basé sur un ensemble d'équations linéaires, dites équations structurelles, qui comme dans les modèles économétriques permettent de rendre compte des relations entre les attributs qui sont d'une part des attributs internes de participation aux activités et d'autre part des variables explicatives extérieures. Ces modèles permettent de rendre compte des relations entre les attributs liés aux activités et ceux liés aux déplacements et permettent de prédire les flux de voyageurs. Ainsi, ils représentant une alternative au modèle à quatre étapes actuel. Toutefois, ils ne permettent pas de traduire les processus de prise de décision à l'échelle de l'individu. De plus, comme tous les modèles agrégés, ils ne permettent pas une analyse spatiale ou temporelle. Finalement, ce type de modèles présente encore un certain nombre des limites relevées dans le modèle à quatre étapes.

1.5. Intégrations des modèles et population synthétique

Le processus d'intégration des modèles dépend essentiellement de la nature agrégée ou désagrégée du modèle.

On distingue principalement trois cas de figure (cf. figure 8). Dans un premier cas, les données d'entrée sont agrégées, les activités et les déplacements sont associés à des groupes d'individus, l'affectation au réseau est également agrégée et on obtient en donnée de sortie le débit routier au niveau des nœuds. Dans un second cas de figure les données en entrée sont désagrégées, un agenda est créer pour chaque individu puis les données de déplacement sont agrégées dans un matrice O-D, l'affectation au réseau est agrégée et on obtient encore une fois en donnée de sortie le débit routier au niveau des nœuds. Enfin, le cas le plus intéressant dans le cadre des modèles basé sur l'activité est celui où les données d'entrée sont désagrégées et utilisées pour générer des agendas individuels qui sont directement utilisés dans un modèle de simulation de trafic multi-agent qui permet d'obtenir en sortie un historique (désagrégé) des événements sur le réseau de transport.

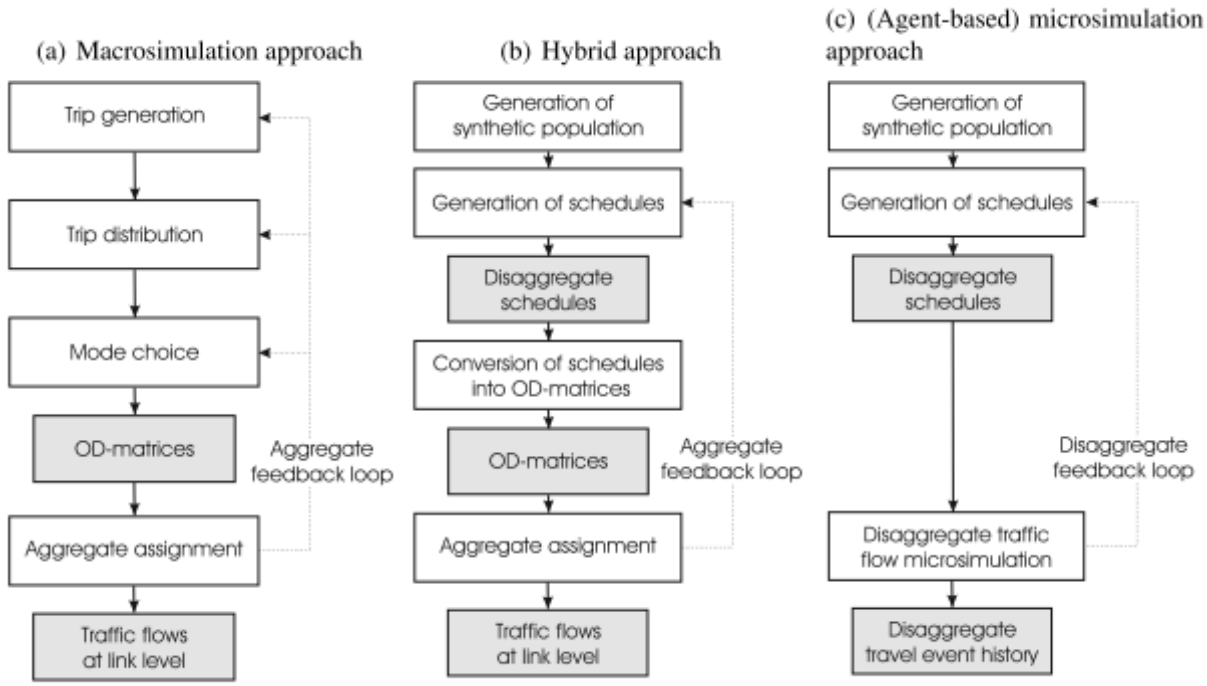


Figure 8 Comparaison de différentes approches pour intégrer les modèles estimant la demande de transport et les modèles d'affectation (source Feil, 2010 p.28)

Dans les modèles basés sur l'activité, l'individu est l'élément central. De plus, dans les modèles désagrégés, chaque individu est considéré individuellement, or, afin de définir un agenda détaillé et réaliste pour chaque personne, il est indispensable de disposer des données d'entrée (attributs) propres à chaque individu.

Avoir une base de données contenant tous les attributs nécessaires à la modélisation pour chacun des individus de la population est un enjeu majeur qui est confronté à plusieurs problèmes.
Il existe principalement deux sources de données : le recensement national et les enquêtes de mobilité des ménages.

Le recensement de la population fourni des informations sur l'ensemble de la population mais présente deux problèmes. Tout d'abord ce n'est pas une base de donnée construite dans le but

d'étudier la demande de transport, en ce sens elle ne renseigne que partiellement sur la mobilité. De plus, les données de recensement doivent conserver l'anonymat afin de garantir le respect de la vie privée. De ce fait, les données relatives à un individu ne peuvent pas être associées à une situation géographique trop précise ; les données spatiales sont donc agrégées à une échelle qui permet de conserver cet anonymat.

Les enquêtes de mobilité des ménages sont plus ciblées et permettent d'obtenir des informations plus précises et plus nombreuses sur la mobilité. Toutefois, les enquêtes sont relativement coûteuses à mener et, de plus, on ne peut forcer les personnes à y répondre. Les données recueillies ne concernent donc qu'un échantillon de la population souvent trop petit pour être statistiquement représentatif.

La solution pour résoudre les lacunes des données disponibles est de passer par la création d'une population synthétique. Une population synthétique est une population artificielle composée d'individus fictifs, alors appelés agents, auxquels sont associés des attributs individuels. Il existe essentiellement deux méthodologies pour créer une population synthétique.

Reconstruction Synthétique (SR)

L'approche la plus répandue est la reconstruction synthétique également appelées méthode du clonage ou méthode de facteurs de grossissement.

Cette façon de procéder a été introduite par Beckman et al. en 1996. Le principe est de recréer, pour chaque secteur du recensement, une population en clonant les attributs d'individus sondés dans ce secteur.

Cette démarche utilise à la fois des données désagrégées issues d'enquêtes et des données agrégées provenant en général du recensement. Il y a deux étapes dans cette méthodologie. La première consiste à établir la table des "proportions" ce que Beckman et al. font en utilisant la méthode d'ajustement proportionnel itératif (ou Iterative Proportional Fitting (IPF) ou encore méthode de redressement par quotient).

L'IPF est un algorithme itératif développé par Deming et Stephan (1940). Cet algorithme permet de transformer les valeurs d'une matrice de sorte que les sommes des colonnes ainsi que les sommes des lignes équivalent à des valeurs cibles.
Notons que cette méthode peut être appliquée à des matrices de dimension supérieure à deux.

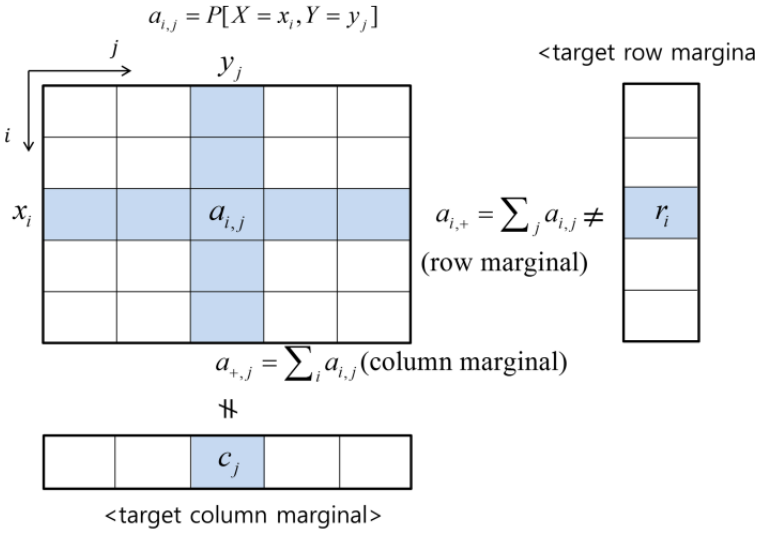


Figure 9 Principe de la méthode IPF (Jeong, Lee, Kim et Shin, 2016)

La seconde étape consiste à créer une population synthétique à partir des types démographiques de l'enquête. Un "clonage" des individus est effectué en suivant la probabilité d'apparition de chaque profil d'individu sondé dans la population réelle; le but étant de faire correspondre les données agrégées de la population réelle avec les données agrégées de la population d'agents.

Cette méthode peut être utilisée en clonant des individus, mais elle peut être utilisée pour cloner des ménages. Beckman et al. eux-mêmes clonent des ménages. Ce qui permet de rendre compte de l'organisation en foyer de la population dont tiennent compte également la plupart des modèles de transport basés sur l'activité. L'enjeu dans les modèles postérieurs est en outre d'assurer une bonne représentation de la population tant à l'échelle des ménages qu'à l'échelle des individus (cf. Guo J.Y. et Bhat C.R. (2007); Arentze T., Timmermans H.J.P. et Hofman F. (2007); Ye X., Konduri K., Pendyala R.M., Sana B., and Waddel P. (2009)).

Dans certains modèles, l'IPF peut être remplacé par une autre méthode statistique notamment par un algorithme de repondération ou une simulation de Monte Carlo.

Les méthodes de reconstruction synthétique de la population présentent certains inconvénients liés en majeure partie à la qualité de l'enquête utilisée. En effet, en fonctionnant par clonage le risque est de multiplier les erreurs d'observation autant de fois que l'on réplique le profil des individus. D'autre part, si certains types d'individus n'ont pas été observés au cours de l'enquête ils demeureront non-représentés.

Hill Climbing (HR) ou Optimisation Combinatoire (CO)

Une autre approche s'inspire des algorithmes d'escalade utilisés en science informatique. Le terme d'algorithme d'escalade fait référence à une technique d'optimisation mathématique ; le principe est de faire évoluer par itérations une solution vers une meilleure solution. L'algorithme converge vers un optimum atteint (avec une erreur négligeable) lorsque deux solutions successives du processus itératif présentent des différences négligeables.

Le principe pour créer une population à partir de cette technique est de sélectionner au hasard des individus (ou des ménages) d'enquêtes.

Si la combinaison d'individus sélectionnés ne correspond pas avec les données de recensement du secteur géographique considéré, un certain nombre d'individus sont remplacés par de nouveaux individus provenant d'enquêtes de sorte que la nouvelle population soit plus proche de la population réelle. Cette étape est répétée jusqu'à atteindre un optimal.

Williamson présente un modèle utilisant une méthode d'optimisation combinatoire pour créer une population synthétique dynamique. Il crée une sélection aléatoire d'individus dont le nombre correspond au nombre d'habitants du secteur. Il compare ensuite les attributs agrégés de la sélection avec les données de recensement. Si la correspondance ne satisfait pas les critères de précision fixés, un individu est remplacé aléatoirement par un nouvel individu de l'enquête. Les attributs de la nouvelle population fictive sont comparés à ceux de la population réelle ; si une amélioration est observée, le remplacement est effectué. Ce processus est répété jusqu'à l'obtention d'une population qui corresponde à la population réelle selon la précision visée ou jusqu'à l'obtention d'une population qui ne puisse pas être améliorée.

Désagrégation spatiales des données

Les modèles présentés ci-dessus permettent de créer une population synthétique d'individus (éventuellement regroupés en ménages) associés à un ensemble de données de mobilité. Toutefois, ces modèles ne permettent pas de résoudre le problème d'agrégation spatiale de la population. Il reste encore à répartir spatialement la population sur l'ensemble du secteur. De même, il faudra être capable d'assigner un lieu de travail aux individus ayant un emploi.

Les données désagrégées qui concernent les lieux de résidence précis et les lieux de travail, ne sont généralement pas disponibles ou sont d'accès limité. Par ailleurs, si les données étaient disponibles pour un échantillon de la population, elles ne seraient pas exploitables de la même

manière que les autres données désagrégées des modèles de création de populations synthétiques (attribuer par copie la même adresse aux différents individus de la population est absurde).

Les façons de procéder dépendent en grande partie des données disponibles pour décrire l'organisation ainsi que la répartition des logements et des emplois, les densités. Un premier type de document utilisable est le plan de secteur (ou plan d'urbanisme) qui existe dans la plupart des pays et qui définit les affectations du sol (industrie, habitat, activité économique, zone agricole, zone forestière, zone de loisirs...). Des cartes topographiques peuvent permettre de localiser la répartition des bâtiments.

Pour la France, la plateforme de modélisation MobiSim utilise une base de données topographique et détermine pour chaque bâtiment un type (résidentiel ou immeuble), un volume et un nombre de pièces puis à chaque logement sont affectés des parkings, des véhicules et un ménage (issu d'une population synthétique). D'autre part, la plateforme utilise la base de données Siren de l'Insee (institut national de statistique et des études économiques) ; ces données sont disponibles en libre accès. La base donnée Siren répertorie pour chaque entreprise, son adresse, son type d'activité et le nombre d'employés. A partir de cette base de données il est possible d'affecter des lieux d'activité et notamment des lieux de travail aux individus de la population synthétique.

Dans une méthode de création de population développée pour les Etats-Unis par Beckman et al. (2015), les modélisateurs utilisent le réseau routier de la base de données HERE (NAVTEQ). La localisation résidentielle est associée à une route et s'appuie sur le type de route (des routes moins large desservent généralement des maisons unifamiliales) et la distribution des types de bâtiment. Les lieux de travail sont associés à chaque individu en suivant une loi de probabilité proportionnelle aux effectifs d'employés et inversement proportionnelle à la distance. La base de données Dun & Bradstreet (D&B) fournit la localisation des entreprises ainsi que le type et le nombre d'emplois.

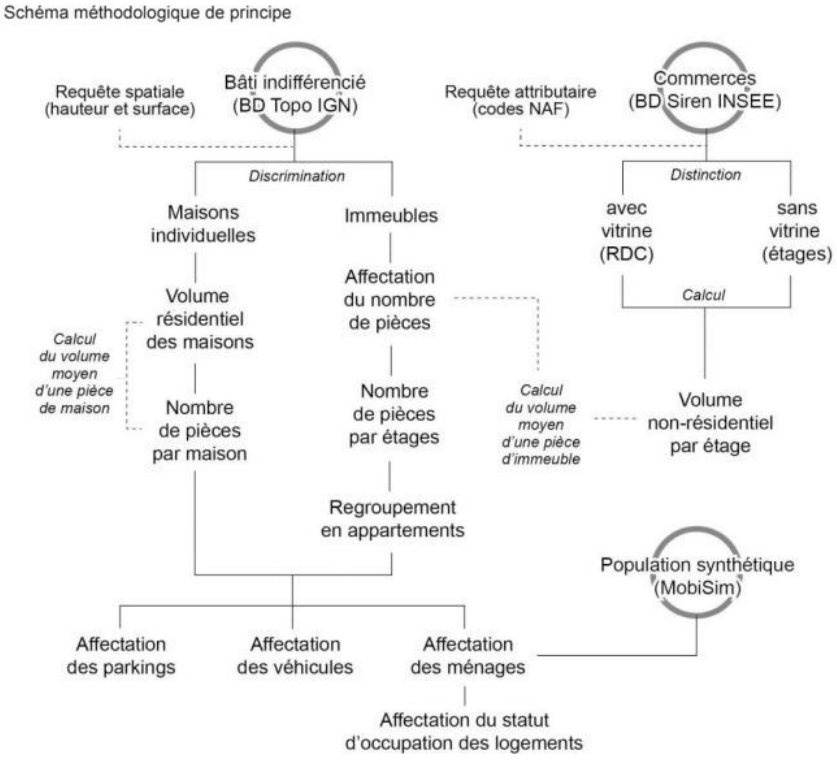


Figure 10 Schéma méthodologique de principe de désagrégation de données spatiales du modèle MobiSim (source: Antoni, Aupet et Vuidel, 2011)

Chapitre 2

L'EXPLOITABILITE DE LA BIOSTATISTIQUE POUR LA MODELISATION DE PHENOMENES DE TRANSPORT

2.1. La biostatistique

Qu'est-ce que la biostatistique ?

La biostatistique se définie simplement. C'est l'application de la science statistique à la biologie et à la médecine.

La biologie est, par définition, la science des êtres vivants. Elle recouvre un très vaste domaine qui va de la biologie cellulaire à la botanique en passant par la biochimie, la zoologie, la bionomie, l'anatomie, la biologie humaine, la paléontologie etc.

Quant à la Statistique, si l'on se réfère à la définition donnée par le dictionnaire Larousse, elle est l' "ensemble des méthodes qui ont pour objet la collecte, le traitement et l'interprétation de données d'observation relatives à un groupe d'individus ou d'unités". Cette définition est très générale.

En fait, face à l'étendue et à la complexité de leur science, les statisticiens ont longtemps débattu sans arriver à un consensus de définition. Néanmoins, une notion centrale est mise en évidence par le Comité National de l'Organisation française qui écrit que "la statistique consiste en une méthode qui, par relevé en masse et le groupement rationnel des faits, permet de décrire et d'observer les phénomènes collectifs, d'obtenir des rapports numériques sensiblement indépendants des anomalies du hasard, de dégager la régularité du changement." La notion essentielle qui apparait dans cette définition à travers les termes de "régularité" et de "changement" est la notion de variabilité. La statistique est la science qui étudie la variabilité.
Or, justement la variabilité est une propriété fondamentale des systèmes biologiques ; c'est donc naturellement que les biologistes vont être amenés à recourir à la statistique.
Bien sûr, la biologie n'est pas le seul domaine d'application de la statistique. Ceux-ci sont nombreux : la démographie, le marketing, la géophysique, l'informatique, l'industrie ... ainsi bien sûr que le domaine de la prévision des phénomènes de transports.

Si on revient à la définition du terme 'biostatistique', la biostatistique semble couvrir un champ très large. En effet, par définition, la biostatistique comprend la biométrie ainsi que le recueil, l'analyse et le traitement statistique de données recueillies lors d'études écologiques, biologiques, agronomiques, halieutiques, etc. Toutefois, en pratique, les applications associées à la biostatistique couvrent un champ bien moins vaste. Les applications concernent, en fait, deux grandes thématiques : d'une part la médecine et la santé (santé publique, épidémiologie, essai clinique, recherche médicale,...) et d'autre part l'étude des gènes (génomique, génétique,...).

C'est bien cette partie de la biostatistique plus liée à l'inférence statistique qui nous intéresse plutôt que le sens large qui est associé à ce mot valise.

Pourquoi s'intéresser à la biostatistique pour décrire des phénomènes de transport ?

Comme évoqué plus tôt la statique est une science qui concerne un grand nombre de domaines d'application. C'est un outil développé par les statisticiens qui est mis au service des chercheurs de tous domaines de la même façon que les mathématiques. Cependant les "outils" utilisés et surtout la façon de les utiliser varient d'un domaine à l'autre. Les problèmes étudiés ainsi que la façon de les aborder mènent à l'élaboration de méthodologies et de techniques différentes.

Il semble donc très intéressant d'avoir une vision transversale. Pierre Dagnelie affirme que "les statisticiens ont tout intérêt à exploiter au maximum la diversité de leur discipline, pour ceux qui ressentent le caractère enrichissant de cette diversité, ou à surmonter les différences qui existent entre eux, pour ceux qui croient se heurter à des barrières " (Diversité et unité de la statistique).

Pourquoi le domaine de la biostatistique en particulier ? D'autres domaines semblent pouvoir également fournir des techniques exploitables pour la modélisation de phénomènes de transport. On peut par exemple penser au domaine des sciences sociales ou au marketing qui utilisent les statistiques afin de décrire et prévoir les comportements humains.

Dans le contexte actuel du domaine d'étude du transport et de la mobilité urbaine, où les chercheurs créent des modèles basés sur les séquences d'activités des populations, la biostatistique, et en particulier la statistique génétique qui étudie des séquences d'ADN et de protéines, est une source prometteuse de méthodes et de techniques à approprier en transport.

***

Mon but ne sera pas ici de faire un état de l'art exhaustif de la biostatistique mais de décrire des techniques biostatistiques pertinentes qui semblent représenter une nouvelle ressource pour développer les modèles de transport.

Afin de mener mes recherches dans le domaine de la biostatistique, je me suis appuyé sur mon travail de recherche d'état de l'art en transport qui est délibérément et par nécessité le plus exhaustif possible.

Mes investigations m'ont plutôt conduit vers la statistique génétique et vers la bioinformatique. Toutefois, je ne prétends pas que mon travail soit exhaustif. Des méthodes utilisées en santé publique par exemple pourraient également être considérées.

2.2. Des modèles et des techniques à transposer

2.2.1. Alignement de séquences

L'alignement de séquences est une technique fondamentale de la bioinformatique. Elle consiste à identifier des séries de caractères similaires qui apparaissent dans le même ordre dans deux séquences (alignement par paire) ou dans plusieurs séquences (alignement multiple) biologiques (ADN, ARN ou protéines).

L'enjeu est de déterminer une chaîne de modifications qui permet de passer d'une séquence à l'autre. Les modifications mises en jeu traduisent des mécanismes à l'origine des mutations et de l'évolution des espèces. Ces mécanismes sont la substitution, le remplacement d'un gène par un autre, l'insertion et la délétion. La délétion et l'insertion font apparaître des trous dans les chaînes.
Faire apparaître les "gaps" associés aux insertions et aux délétions est le mécanisme qui permet d'aligner les séquences qui sont généralement de longueurs différentes.

Les objectifs de l'alignement de séquences sont multiples. Il permet notamment d'établir la distance et le lien de parenté entre séquences, ou d'identifier des zones de concordance dans les séquences et de les relier à des fonctionnalités.

Les séquences qui sont très semblables, ou "similaires", ont probablement une même fonction. De plus, si deux séquences qui proviennent de deux organismes différents sont similaires, alors il est fort probable qu'il existe un "ancêtre commun" à ces séquences. On parle alors de séquences homologues. L'alignement de séquence est très utile en phylogénie, l'étude des relations de parenté entre êtres vivants.

On distingue deux grandes catégories d'algorithmes d'alignement de séquences. Ceux faits pour comparer deux séquences entre elles et ceux faits pour comparer des ensembles contenant trois séquences ou plus. L'alignement de paires de séquences est plus simple à mettre en place dans un premier temps. Notamment, parce que pour ce type d'algorithmes, les problèmes de complexité algorithmique ne se posent pas ; ce qui permet d'imaginer des algorithmes proposant des méthodes optimales. Cependant, il peut être également intéressant de comparer entre elles un plus grand nombre de séquences homologues ; on parle alors d'alignement séquentiel multiple. Entre autre les méthodes d'alignement multiple sont utiles pour l'approche évolutionniste et la création d'arbres phylogéniques qui font apparaître les relations de parenté entre les espèces.

Par ailleurs, il existe deux types d'alignements de séquences : l'alignement global et l'alignement local.

Alignement global

Pour l'alignement global, il s'agit d'aligner deux séquences sur toute leur longueur en faisant correspondre un maximum de caractères.
Les séquences comparées doivent être de tailles semblables et relativement similaires sur toute leur longueur (homogénéité de longueur et de motifs).

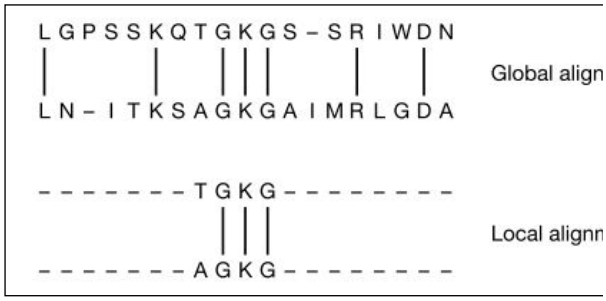


Figure 11 Différence entre alignement global et alignement local

Alignement local

Les méthodes d'alignement local permettent de comparer une séquence avec une (ou plusieurs) partie d'une autre séquence. Ces méthodes sont plus adaptées pour comparer des séquences de tailles très différentes ou qui sont semblables sur certaines sections seulement.

Aperçu des méthodes d'alignement des séquences

Alignement par paires	Global	- Dot Matrix (1970) - Algorithmes de programmation dynamique : algorithme de Needleman-Wunsch (1970)
	Local	- Algorithmes de programmation dynamique : algorithme de Smith-Waterman (1981) - Méthodes par mots ou k-tuples : FASTA (1988), BLAST (1990)
Alignement multiple	Global	- Extension des méthodes d'alignement par paire utilisant des algorithmes de programmation dynamique : MSA (1989) - Méthodes progressives : CLUSTALW (1996), PILEUP (1997) - Méthodes itératives : algorithmes génétiques
	Local	- recherche de motifs : MOTIF (1990), ASSET (1994)

Méthodes par matrices de pixels (Dot Matrix ou dot-plot)

Le principe de la méthode est relativement simple. Cela consiste à écrire deux séquences dans un matrice l'une à l'horizontale et l'autre à la verticale et à représenter par un "point" les positions identiques. Les diagonales de points traduisent alors des motifs identiques dans les séquences.

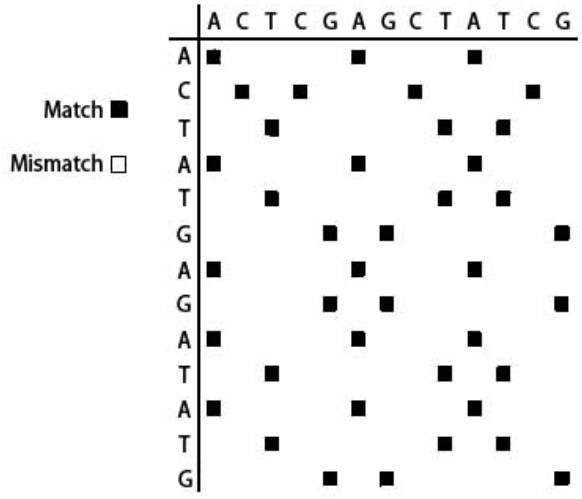


Figure 12 Dot-plot cadre = 1

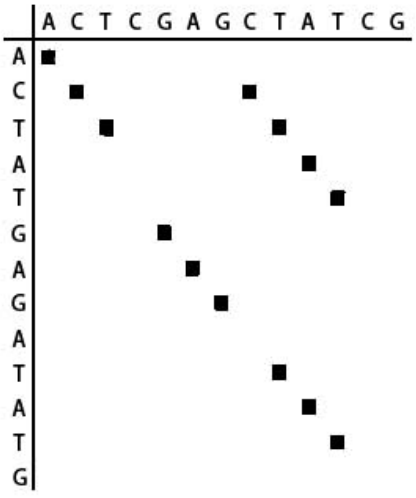


Figure 13 Dot-plot cadre = 3

Le premier avantage de cette méthode est de mettre en évidence de façon claire le degré de similarité entre deux séquences notamment lorsque l'on en n'a aucune idée a priori. Un second avantage est que cette méthode fait apparaître toutes les correspondances et non pas seulement un optimal. La limite de cette méthode est qu'elle ne donne pas concrètement un alignement effectif.

Alignement global : l'Algorithme de Needleman-Wunsch

La méthode est basée autour de la construction d'une matrice de scores de similarité. La matrice est complétée selon un ensemble de règles. Le principe est d'affecter des poids aux différentes opérations de substitution. Ces poids représentent un coût de modification pour passer d'une chaine à l'autre.

Les coûts de modification sont représentés dans une matrice de substitution, accompagnée en général d'une indication du coût des gaps (insertion/délétion). Par exemple, pour l'ADN, on définit communément : match = +2, mismatch = -1, gap = -2. Pour les protéines il existe des matrices de références telles que PAM et BLOSUM.

Le but est de déterminer dans la matrice de scores, le chemin le plus "court", c'est à dire celui qui maximise le score. Pour cela, on procède de proche en proche en partant de la première case de la matrice et en avançant sur la case contigüe qui présente le meilleur score.
Le chemin détermine l'alignement optimal et s'interprète de la manière suivante :

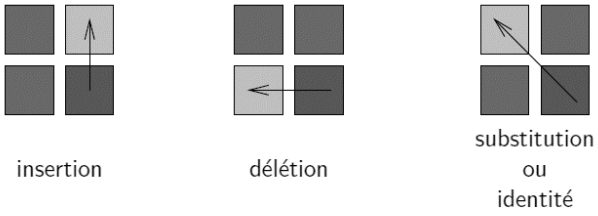


Figure 14 Exemple de mise en œuvre de l'Algorithme de Needleman-Wunsch

L'algorithme de Needleman-Wunsch agit de la même manière quelle que soit la longueur ou la complexité des séquences et garantit de trouver le meilleur alignement possible, pour une matrice de substitution donnée. La difficulté de la méthode est de définir une matrice de coûts pertinente.

Alignement des chaînes suivantes : Chaîne 1 : ACGGCTAT ; Chaîne 2 : ACTGTAG

Matrice de substitution

	A	C	G	T
A	+2	-1	-1	-1
C	-1	+2	-1	-1
G	-1	-1	+2	-1
T	-1	-1	-1	+2

Le coût d'un gap est de -2.

Matrice de scores de similarité

	--	A	C	G	G	C	T	A	T
--	0	-2	-4	-6	-8	-10	-12	-14	-16
A	-2	2	0	-2	-4	-6	-8	-10	-12
C	-4	0	4	2	0	-2	-4	-6	-8
T	-6	-2	2	3	1	-1	0	-2	-4
G	-8	-4	0	4	5	3	1	-1	-3
T	-10	-6	-2	2	3	4	5	3	1
A	-12	-8	-4	0	1	2	3	7	5
G	-14	-10	-6	-2	-1	0	4	5	6

La matrice de similarité suit la formule de récurrence suivante :

$$
\begin{aligned}
\mathtt{Sim}(0,0) &= 0 \\
\mathtt{Sim}(0,j) &= \mathtt{Sim}(0,j-1) + \mathit{Ins}(V(j)) \\
\mathtt{Sim}(i,0) &= \mathtt{Sim}(i-1,0) + \mathit{Del}(U(i)) \\
\mathtt{Sim}(i,j) &= \max \begin{cases} \mathtt{Sim}(i-1,j-1) + \mathit{Sub}(U(i),V(j)) \\ \mathtt{Sim}(i-1,j) + \mathit{Del}(U(i)) \\ \mathtt{Sim}(i,j-1) + \mathit{Ins}(V(j)) \end{cases}
\end{aligned}
$$

>Alignement Global :

A	C	G	G	C	T	A	T
A	C	T	G	--	T	A	G

Alignement local : l'Algorithme de Smith-Waterman

L'algorithme de Smith-Waterman est un algorithme optimal qui donne un alignement correspondant au meilleur score possible de correspondance entre les chaînes comparées.

Le procédé d'alignement de l'algorithme de Smith-Waterman est proche de celui qui est utilisé dans l'algorithme d'alignement global de Needleman-Wunsch. En effet, de la même manière que l'algorithme de Needleman-Wunsch, l'algorithme de Smith-Waterman se base sur une matrice de substitution qui indique les scores de substitution entre les éléments. La différence est que l'objectif n'est plus de comparer deux séquences dans leur intégralité mais de rechercher des régions ou segment particulièrement similaires.

Ainsi la différence est que la valeur d'un "gap" est désormais une variable qui dépend de la longueur du gap. La matrice de scores est remplie de manière quasi analogue à ce qui est proposé dans l'algorithme de Needleman-Wunsch, sauf que l'on ne prend en compte que les valeurs positives, permettant en quelques sortes à la chaîne "de se couper" en repartant à zéro dès qu'une séquence se termine.

Une fois la matrice de scores complétée, il s'agit simplement de trouver le score maximal atteint dans la matrice et de remonter le chemin jusqu'à rencontrer un zéro : ce chemin correspond au meilleur alignement local entre les deux chaînes.

FASTA et BLAST

FASTA et BLAST sont des algorithmes qui utilisent une méthode par mots ou k-tuplets. Le principe est de rechercher des motifs (suites de caractères) relativement courts puis d'utiliser ces "mots" pour procéder à l'alignement en utilisant une méthode de programmation dynamique locale.
Ces méthodes sont des méthodes heuristiques qui présentent l'avantage d'être rapides (temps de calcul réduit) mais qui demandent de porter une attention particulière à la pertinence des résultats obtenus.

Extension des méthodes de programmation dynamique : MSA (Lipman et al. 1989)

Concernant les alignements multiples, une première façon de procéder est de se baser sur les méthodes de programmation dynamique utilisées pour l'alignement par paire. Les algorithmes de Needleman-Wunsch ou de Smith-Waterman sont des exemples de méthodes de programmation dynamique.

Les méthodes de programmation dynamiques construisent une matrice de scores dans laquelle chaque position fournit le meilleur alignement possible au fur et à mesure de sa construction. On peut étendre cette analyse à trois ou plus de trois séquences en ajoutant des dimensions à la matrice : pour trois séquences on obtient alors une sorte de cube en treillis. De la même manière que précédemment on remplit les scores correspondant à chacune des positions du treillis en trois dimensions.

Cette méthode peut fonctionner, cependant elle est très lourde en termes informatiques puisqu'elle présente une complexité qui varie exponentiellement avec le nombre de séquences comparées. Ainsi, par cette méthode, seul un petit nombre de séquences courtes pourront être comparées.

Méthodes progressives : CLUSTALW et PILEUP

L'alignement multiple d'un ensemble de séquences peut aussi être vu comme une histoire évolutive des séquences. Si les séquences s'alignent bien, alors elles sont probablement dérivées d'un même ancêtre commun. Cette idée est exploitée dans les méthodes progressives.

Dans un algorithme progressif tel que ClustalW de Higgins, les séquences sont d'abord toutes comparées par paires, selon des méthodes globales ou locales non optimales ; puis rangées dans une matrice de distances résumant les distances entre toutes les séquences comparées deux à deux. La distance entre deux séquences est donnée par la formule suivante :

$$Distance\ entre\ deux\ séquences = 1 - \frac{Nombre\ de\ résidus\ identiques}{Nombre\ de\ résidus\ comparés}$$

Ensuite, on construit un arbre guide ou arbre phylogénique. Il existe de nombreuses façons de construire un arbre guide à partir d'un matrice de distance ; CLUSTALW utilise la méthode dite « neighbourjoining ». Pour cela, on suit les étapes suivantes :

1 – on joint les deux séquences les plus proches dans la matrice
2 – on crée un nouveau nœud qui est le plus proche ancêtre commun des nœuds joints.
3 – on calcule la distance des nœuds de la paire avec leur ancêtre commun
4 – on calcule la distance des autres nœuds avec l'ancêtre commun
5 – on recommence l'algorithme en considérant la paire de voisins comme un seul nœud remplacé par leur ancêtre commun.

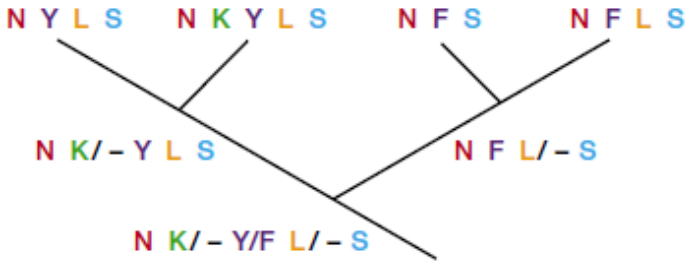


Figure 15 Alignement de séquence progressif - arbre guide ou arbre phylogénique : les séquences les plus proches sont liées à un même nœud sur lequel se trouve leur ancêtre commun

Enfin, on réalise l'alignement progressif en suivant l'ordre donné par l'arbre guide. Ainsi, les séquences les plus proches sont alignées en premier, puis des séquences ou groupes de séquences additionnels sont ajoutées, suivant les alignements initiaux un alignement multiple faisant apparaître dans chaque colonne les variations de séquences pour passer d'une séquence à une autre.

PILEUP est très similaire à CLUSTALW. Les séquences sont alignées par paire en utilisant l'algorithme de Needleman-Wunch, et les scores sont utilisés pour construire un arbre par la méthode nommée « unwaited pair-group method ». Cet arbre est ensuite utilisé pour guider l'alignement des séquences de proche en proche. Contrairement à CLUSTALW cet algorithme n'a pas eu d'amélioration récente.

Le problème des alignements progressifs est qu'ils dépendent fortement des alignements de séquences par paires. Les premières séquences à être alignées sont celles qui sont les plus proches dans l'arbre guide ; si ces séquences sont très proches les résultats obtenus devraient être bons, en revanche, si ces séquences sont éloignées, d'avantages d'erreurs vont apparaître et ces erreurs vont se propager dans toute la méthode d'alignement multiple.

Méthodes itératives : algorithmes génétiques

Les méthodes itératives tentent d'apporter une manière de contrer l'effet négatif rencontré par les méthodes progressives, à savoir la propagation des erreurs dans l'ensemble de l'application de la méthode. Pour cela, on forme de manière répétitive des sous-groupes de séquences que l'on aligne dans un alignement de séquence global comprenant toutes les séquences. La sélection de ces groupes peut se faire selon un arbre phylogénique prédit de la

même manière que dans les méthodes progressives vues précédemment, mais elle peut aussi se faire de manière tout à fait aléatoire.

L'algorithme génétique est un algorithme général du domaine informatique. Il a été adapté pour l'alignement de séquences multiples à partir des années 1997 (Notredame et Higgins). Le principe de cette méthode est d'essayer de générer plusieurs alignements multiples de séquences par des réarrangements qui simulent l'insertion de gap et la recombinaison d'événements au cours de la réplication dans le but de générer un score d'alignement de plus en plus élevé. Ainsi, on commence par générer une centaine d'alignements multiples possibles sur les séquences, puis ces alignements sont associés à un score selon la méthode des sommes de paires (SSP), le meilleur score SSP est le plus bas. Ces alignements multiples initiaux sont ensuite répliqués pour donner une nouvelle génération d'alignements multiples. Une moitié constituée de ceux qui ont les meilleurs scores est envoyée dans la génération suivante sans qu'ils soient modifiés, les autres subissent des mutations. Le processus de scoring, de réplication et de mutation est généralement répété plus de cent fois avant de garder le meilleur alignement multiple.

Méthodes locales : recherche de motifs et de blocks : MOTIF (1990) et ASSET (1994)

Les blocks représentent une région qui se conserve dans un alignement multiple. On peut chercher à trouver des similarités locales dans des séquences sans produire un alignement global. Dans de telles méthodes les séquences sont analysées par recherche de motifs ou par des méthodes statistiques.

Des analyses de recherche de motifs ont été menées sur des groupes de protéines liées, et les motifs d'acides aminés qui ont été localisés peuvent se trouver dans le catalogue Prosite (Bairoch 1991). Ce catalogue regroupe des protéines qui ont des fonctions biochimiques similaires en se fiant à la base de motifs d'acides aminés tels que ceux du site actif. Par conséquent, ces familles ont été utilisées pour la recherche de motifs d'acides aminés par le programme MOTIF (Smith et al. 1990) ; on a trouvé des motifs du type aa1 d1 aa2 d2 aa3, où aa1 et aa2 sont les acides aminés conservés et, d1 et d2 sont des brins d'une longue séquence allant jusqu'à 24 acides aminés. Ces motifs initiaux d'une longueur comprise entre 3 et 60 acides aminés se trouvent dans la base de données BLOCKS. Cependant la taille des motifs du programme MOTIF est limitée. De plus le programme MOTIF fournit toujours un motif, même pour des séquences aléatoires, ce qui remet en question le résultat fournit par le programme : rien ne garantit que le motif fournit soit pertinent. Un algorithme rigoureux nommé ASSET (Aligned Segment Statistical Evaluation Tool) a été créé pour trouver des motifs dans des séquences comptant jusqu'à 50 acides aminés, les regrouper, et fournir une mesure de la pertinence de ces motifs.

Intérêt des techniques d'alignement de séquence pour la prévision des phénomènes de transport

Des chercheurs se sont déjà intéressés à l'utilisation des travaux effectués dans le domaine de la bio-informatique sur les alignements de séquences pour étudier les chaînes d'activités.

Notamment, W. C. Wilson a utilisé l'algorithme CLUSTAL pour étudier des chaînes d'activités. En effet, il a trouvé qu'il existe une certaine similarité entre les problèmes d'analyse de séquences dans le domaine de la biologie moléculaire et les études sur l'utilisation du temps. Il a donc cherché à utiliser les concepts et le vocabulaire des alignements de séquences pour l'analyse de chaînes d'activités.

Dans les chaînes d'activités, chaque activité est codée par une lettre ou un ensemble de lettres. Il existe donc une étape préliminaire importante qui consiste à coder un agenda composé

d'activités humaines par une séquence de lettres. La première question qui se pose concerne le niveau de détail qu'il est raisonnable ou nécessaire de conserver. Par exemple, Wilson indique qu'il a décidé de coder séparément « boire un café » et « boire un thé » mais qu'il a codé « prendre un café » comme « boire un café ».

L'algorithme CLUSTAL emploie des scores de similarité, or les scores entre acides aminés ne sont pas applicables directement aux activités. Il faut donc les définir pour toutes les activités qui entrent dans les chaînes utilisées. Ces scores permettront la substitution d'activités relativement similaires et pénaliseront la substitution d'activités très distinctes et non substituables.

Table 1. Classification of nineteen activities.

sleep and rest (r)	eating and drinking (e)	full-time education (f)
private leisure (l)	watching television (q)	sport (a)
travel, public modes (g)	community service, religious observation, organizations (n)	organized leisure, entertainment (m)
casual social (v)	employment, main or second job (w)	child, family care (k)
domestic work (d)	travel, private modes (t)	cooking and washing dishes (c)
personal care, hygiene (p)	shopping, use of services (s)	unknown activity (y)
information, reading, study (i)		

Mrs Yeats's daily activities expressed as character sequences are as follows:

Wednesday	rcedetst0celpprceqvcr
Thursday	rpeddsed0icecgsgcecqvr
Friday	rpeciddptnt0cecitstdlcecqldepr
Saturday	rpecddp0cecilvcecyptvmtpr
Sunday	rpecddptn0tcrcecicectntqecqr
Monday	rcecddtstdc0eecidlcecqtitqeqr
Tuesday	rcecddd0ctitcecpgvtvpqtitveqr

Figure 16 Exemple d'un ensemble d'activités et de chaînes d'activités, issu de (Wilson, 1998)

Wilson applique l'algorithme CLUSTAL à plusieurs chaînes d'activités d'une même personne sur des journées différentes, ou sur des groupes de personnes distinctes, et observe la nature des résultats et les observations que l'on peut en faire. Notamment on peut faire apparaître des motifs qui se répètent chez la même personne d'une journée à l'autre, faisant ressortir des habitudes individuelles, ou des motifs entre les chaînes d'activités de plusieurs personnes faisant apparaître des points communs entre elles. Plus l'ensemble de personnes sera grand et plus les points communs seront durs à trouver.

Noam Shoval and Michal Isaacson ont également mené des travaux dans cette optique. Ils expliquent comment les travaux d'analyse de séquences menés en bioinformatique peuvent être utilisés dans les recherches sur les activités humaines et l'allocation de temps et de moyens de transports. Ils expliquent que l'on peut utiliser l'analyse de séquences de deux manières distinctes : soit pour produire des groupes selon leurs motifs d'activités, soit pour détecter des modèles de comportement dans les séquences étudiées.

Pour leur travail les auteurs ont utilisé une forme de l'algorithme Clustal adaptée pour répondre à des problèmes de géographie. Ils nomment ce nouvel algorithme ClustalG. Les espaces géographiques sont découpés en polygones, chaque polygone est représenté par un caractère. La dimension temporelle est représentée par le fait que chaque séquence est codée en utilisant le même pas temporel entre deux positions géographiques.

Ils utilisent ensuite ClustalG pour analyser le comportement de 50 visiteurs dans Akko, une ville du Nord de l'Israël. L'implémentation de l'alignement de séquence dans ce cas n'est pas utilisée sur des séquences d'activités mais sur des séquences de positions. L'étude est menée avec une résolution de 1 min, ainsi chaque caractère représente une minute. Par ces études les auteurs ont pu identifier trois groupes distincts de visiteurs selon les parcours qu'ils ont suivis. Leur étude démontre que l'utilisation d'algorithmes d'alignement de séquences dans le domaine de l'analyse des activités humaines est pertinente.

En résumé, les outils d'alignement de séquences permettent d'"aligner" des séquences (de symboles) de longueurs égales ou inégales. Ainsi, les techniques d'alignement de séquences peuvent permettre de mettre en évidences des activités communes entre différents agendas et de faire apparaître des motifs d'activités récurrents. D'autre part, elles permettent également d'établir un score d'alignement qui traduit la plus ou moins grande similitude entre deux séquences.

Les techniques d'alignement de séquences sont donc un outil de choix pour l'analyse de séquences d'activités. Elles peuvent permettre de détecter des activités quotidiennes et des activités hebdomadaires ou plus ponctuelles (Wilson, 1998). Elles peuvent être utilisées pour mettre en évidence des motifs récurrents ou détecter des modèles de comportement. Elles peuvent également servir à établir une classification des agendas, et pour aller plus loin sur ce sujet, on pourra s'intéresser notamment au travaux menés dans le domaine de la phylogénie qui étudie les "parentés" entre les organismes vivants et établie une classification organisée sous la forme d' "arbres de vie".

D'autre part, les outils de comparaison de séquences qui permettent de comparer (notion de score d'alignement) des séquences homogènes de longueurs inégales, peuvent servir pour comparer des séquences d'activités ou des agendas. En outre il existe certains travaux exploitants des outils de comparaison de séquences pour certaines applications dans des modèles de transport basés sur l'activité. On peut notamment citer le travail de Sammour et al. (2012) sur l'utilisation de méthodes d'alignement de séquences pour effectuer une validation dans des modèles basés sur des règles ou encore la publication de Joh, Arentze et Timmermans (1999) qui traite de l'utilité des méthodes d'alignements multiples pour l'analyse de motifs d'activités.

En revanche, si les méthodes d'alignement trouvent tout leur intérêt pour l'analyse et la comparaison de séquences d'activités, leur utilisation ne sera forcément pertinente pour la comparaison de profils d'individus. En effet, les profils (i.e. les caractéristiques utiles dans le cadre de la modélisation du besoin de transport) d'individus sont des séquences de données inhomogènes et de longueur toujours égale (correspondant au nombre d'attributs pris en compte), ainsi, ces séquences n'ont pas besoin d'être alignées, puisqu'elles le sont déjà, et il existe des façons d'établir des scores de similitudes et des méthodes de classification plus pertinentes et plus efficaces que de passer par l'utilisation de techniques d'alignement de séquences.

2.2.2. Analyse de séquences nucléiques

L'analyse de séquences est à la base du travail des chercheurs dans le domaine de la génétique et de la génomique. Il existe dans le « vivant » différents types de macromolécules définissant un code qui permet d'exécuter des processus cellulaires et moléculaires.

Les gènes sont transcrits en ARN messagers qui sont à leur tour traduits en protéines. Ce sont les protéines, chaînes d'acides aminés, qui sont les principaux agents des processus biologiques de la cellule. L'ADN est présent dans toutes les cellules d'un individu sous forme de chaînes orientées qui forment les chromosomes. L'ADN est composé par une succession de quatre molécules appelées bases ou nucléotides : l'adéine (A), la guanine (G), la thymine (T) et la cytosine (C). Ainsi, l'ADN est représenté par les biologistes comme une séquence de lettres dans un alphabet de quatre lettres. Un gène est une portion d'ADN qui code pour une (ou plusieurs) protéine. L'ARN est également une séquence de nucléotides mais composée d'adéine (A), de guanine (G), de cytosine (C) et d'uracile (U).

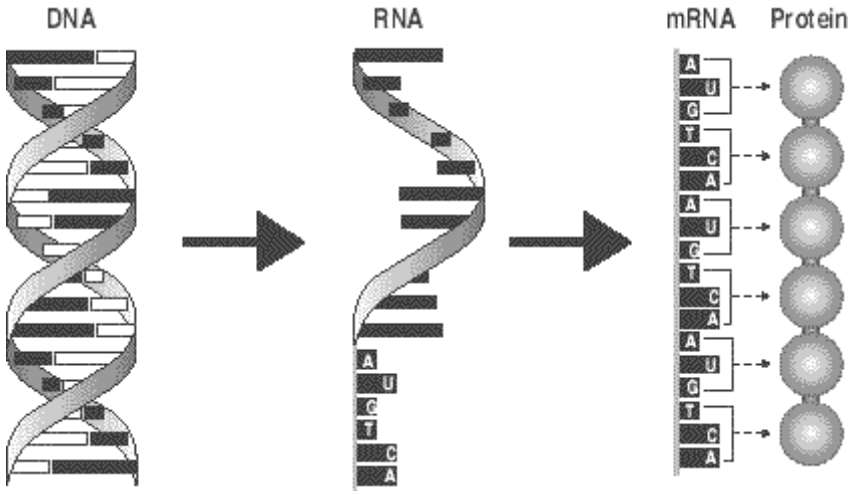


Figure 17 Un gène, un ARN , une protéine (sources thebody.com)

Un des principaux travaux des biologistes est l'analyse des séquences nucléiques de l'ADN. Il s'agit dans un premier temps de localiser les régions codantes (i.e. les gènes) et dans un second temps de déterminer leurs fonctions biologiques. Pour ce faire, les chercheurs recourent à l'utilisation d'outils algorithmiques sophistiqués développés dans le cadre des sciences bioinformatiques.

2.2.2.1. Annotation du génome

Prédiction de gènes

Le premier problème de l'analyse de séquences génomiques est l'identification des régions codantes de l'ADN : il s'agit de repérer les différents gènes (début, exons, fin) sachant que seule une proportion d'environ 5% du génome est codante dans le cas des êtres humains. La proportion de régions codantes varie selon les espèces. On distingue en particulier les eucaryotes des procaryotes ; chez les procaryotes (organismes dont les cellules n'ont pas de noyaux), il n'y a presque pas de régions non-codantes. Chez les eucaryotes, les régions non-codantes sont plus nombreuses et les gènes sont découpés en introns et en exons (partie codante).

Il existe différentes méthodes pour localiser les gènes, et des méthodologies différentes ont été développées selon l'origine de la séquence (eucaryote ou procaryote).

Les techniques de prédiction de gènes reposent sur différents types d'informations (Laurent Duret, 2011) :

1- caractérisation de la taille et du contenu des régions (codantes/non-codantes)
2- caractérisation des signaux au niveau de sites fonctionnels (ex. signaux d'épissage, début et fin de traduction, ...)
3- données expérimentales : transcriptome (protéome)
4- conservation des régions fonctionnelles au cours de l'évolution

Il est possible d'identifier trois grandes techniques de prédiction de gènes :

- Méthodes intrinsèques (ab initio) (utilisent 1 et 2)
- Prédictions par analyse du transcriptome – comparaison de chaîne d'acides aminés (utilisent 3 et éventuellement 2)
- Prédictions par approche comparative – recherche d'exons (ou autres motifs) connus (utilisent 4, et éventuellement 2)

L'ADN et en particulier les parties codantes de l'ADN ne forment pas des séquences aléatoires mais forment un code « organisé ». Les séquences de nucléotides qui constituent l'ADN codent pour la structure des protéines. Les protéines sont des chaînes formées d'une diversité de 22 acides aminés. Chacun des acides aminés est déterminé par un triplet de nucléotides appelé codon. Par ailleurs, il n'y a pas une bijection entre l'ensemble des acides aminés et celui des codons : plusieurs codons peuvent coder pour un même acide aminé. Toutefois, selon les espèces les fréquences d'apparition des codons codant pour un même acide aminé ne sont généralement pas égales ; certains codons sont préférentiellement utilisés.

Table 8.3. *Codon usage table*

UUU-Phe	16.6	26.0	UCU-Ser	14.5	23.6	UAU-Tyr	12.1	18.8	UGU-Cys	9.7	8.0
UUC-Leu	20.7	18.2	UCC-Ser	17.7	14.2	UAC-Tyr	16.3	14.7	UGC-Cys	12.4	4.7
UUA-Leu	7.0	26.3	UCA-Ser	11.4	18.8	UAA-TER	0.7	1.0	UGA-TER	1.3	0.6
UUG-Leu	12.0	27.1	UCG-Ser	4.5	8.6	UAG-TER	0.5	0.5	UGG-Trp	13.0	10.3
CUU-Leu	12.4	12.2	CCU-Pro	17.2	13.6	CAU-His	10.1	13.7	CGU-Arg	4.7	6.5
CUC-Leu	19.3	5.4	CCC-Pro	20.3	6.8	CAC-His	14.9	7.8	CGC-Arg	11.0	2.6
CUA-Leu	6.8	13.4	CCA-Pro	16.5	18.2	CAA-Gln	11.8	27.5	CGA-Arg	6.2	3.0
CUG-Leu	40.0	10.4	CCG-Pro	7.1	5.3	CAG-Gln	34.4	12.2	CGG-Arg	11.6	1.7
AUU-Ile	15.7	30.2	ACU-Thr	12.7	20.2	AAU-Asn	16.8	36.0	AGU-Ser	11.7	14.2
AUC-Ile	22.3	17.1	ACC-Thr	19.9	12.6	AAC-Asn	20.2	24.9	AGC-Ser	19.3	9.7
AUA-Ile	7.0	17.8	ACA-Thr	14.7	17.7	AAA-Lys	23.6	42.1	AGA-Arg	11.2	21.3
AUG-MET	22.2	20.9	ACG-Thr	6.4	8.0	AAG-Lys	33.2	30.8	AGG-Arg	11.1	9.3
GUU-Val	10.7	22.0	GCU-Ala	18.4	21.1	GAU-Asp	22.2	37.8	GGU-Gly	10.9	23.9
GUC-Val	14.8	11.6	GCC-Ala	28.6	12.6	GAC-Asp	26.5	20.4	GGC-Gly	23.1	9.7
GUA-Val	6.8	11.7	GCA-Ala	15.6	16.2	GAA-Glu	28.6	45.9	GGA-Gly	16.4	10.9
GUG-Val	29.3	10.7	GCG-Ala	7.7	6.1	GAG-Glu	40.6	19.1	GGG-Gly	16.5	6.0

Shown are frequency of each codon per 100,000 codons obtained from http://www.kazusa.or.jp/codon/ for *Homo sapiens*; columns 2, 5, 8, and 11, and for *Saccharomyces cerevisiae*, columns 3, 6, 9, and 12.

Figure 18 Codon usage table for Homo sapiens (Source: Mount, 2001, p. 343)

Il existe également des codons qui jouent un rôle particulier : les codons d'initialisation et les codons stop. Ces codons signalent le début et la fin du message génétique sur un ARNm.

Une fois un ORF repéré, il s'agit de déterminer s'il s'agit d'une région codante de l'ADN ou non. Nous avons évoqué précédemment le fait que la structure de l'ADN et en particulier celle des régions codantes n'est pas aléatoire ; ce n'est pas une chaîne aléatoire de codons mais une liste organisée dont la structure est régie par plusieurs facteurs liés à des contraintes en relation avec l'expression du gène et liés à l'origine du gène (facteurs hérités).

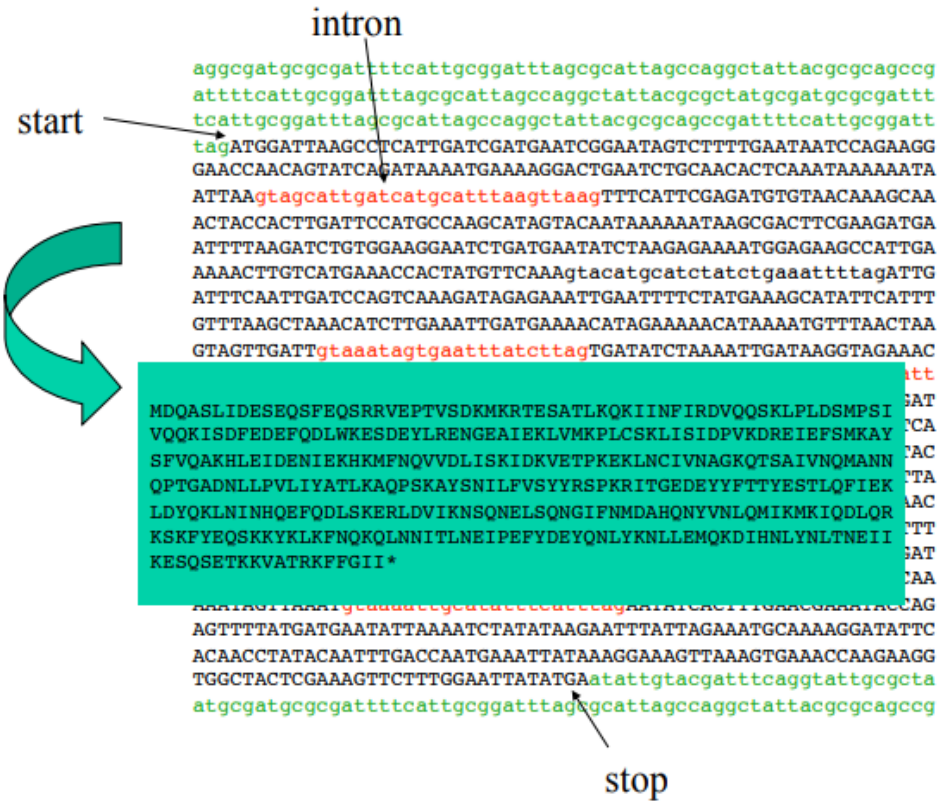


Figure 19 Structure d'un ORF (eucaryotes)

A partir de ce constat que les gènes ont une forme "organisée", trois tests ont été définis afin de vérifier le caractère codant éventuel d'un ORF. Le premier test se base sur le constat que toutes les troisièmes bases tendent à être la même (Fickett 1982). Le second test analyse la fréquence des codons et vérifie si la répartition de codons dans l'ORF est la même que celle trouvée dans les autres gènes d'individus de cette espèce (Staden et McLachlan, 1982). Cela demande de disposer d'une table telle que celle présentée à la figure 18. Le troisième test consiste à traduire l'ADN en une séquence d'acides animés (système de conversion universel) et à comparer cette séquence à une base de données de séquences de protéines existantes. Si une ou plusieurs séquences présentent de fortes similitudes avec la séquence comparée, alors il est probable que l'ORF soit codant (Gish et States 1993). (D'après David W. Mount, p.342-343)

Ainsi, pour localiser les gènes, il existe trois méthodologies complémentaires. Il est possible de recourir à des méthodes de recherche de motifs en identifiant les codons de démarrage et de fin, mais aussi des jonctions entre les introns et les exons, des séquences promotrices, terminatrices, des sites de fixation du ribosome (RBS), voire même encore des exons connus. D'autre part, il est possible de recourir à des méthodes statistiques qui identifient les régions codantes sur la base de l'analyse de la fréquence des codons. Enfin, il est possible de traduire des séquences d'ADN en séquences d'acides aminés et de procéder par comparaison de séquences de protéines.

Afin de déterminer les régions codantes d'un génome, les biologistes ont élaboré une multitudes d'outils qui mettent en application divers modèles statistiques : réseaux de neurones (pour retrouver des gènes complets à partir d'exons connus(ex. Grail II, Uberbacher et Mural, 1991 ; Uberbacher et al. 1996)), discrimination de motifs (utilise des méthodes statistiques pour classifier les ORF d'une séquence sur base de motifs de séquences observés(Solovyev et al., 1994 ; Zhang, 1997), recours à des modèles de Markov cachés.

Prédiction de gènes - Bilan et exploitabilité

Les méthodes employées pour la prédiction de gènes se basent sur la forme même qu'adopte la représentation simplifiée de l'ADN : c'est une séquence de caractères dans un alphabet de quatre lettres. Ainsi, l'ADN est comparable à un texte qui répond à certaines logiques structurantes étant un code qui fait sens.
Partant de ce constat, les généticiens recourent à des outils qui permettent d'une part, de faire une analyse intrinsèque de la structure de l'ADN, et d'autre part, de prévoir les protéines (chaînes d'acides aminés) synthétisées par des ORF supposés codants afin de les comparer aux protéines connues recensées dans des bases de données.

Les agendas peuvent être représentés de diverses manières, néanmoins une façon des plus simples de représenter une chaine d'activités (à la base de tout agenda) est de la représenter comme une suite de lettres où chaque lettre correspond à un type d'activité. Ainsi, une séquence d'activités peut se représenter comme un texte (certes relativement court) écrit avec un alphabet comprenant un nombre fini de lettres (généralement entre 10 et 20 lettres). Ainsi certains des outils permettant de faire une analyse intrinsèque de l'ADN (fréquences des nucléotides, fréquences des successions,...) pourraient permettre d'élaborer des outils d'analyse des séquences d'activités.
En outre, de la même façon que les biologistes testent la vraisemblance qu'une portion d'ADN soit codante, il est envisageable de tester la vraisemblance d'agendas.

Dans ce contexte, les travaux de Rumelhard (2006) concernant la modélisation de l'ADN à l'aide des chaînes de Markov peuvent fournir une base de réflexion intéressante.

Fonction des gènes

Une fois les gènes identifiés, les chercheurs doivent prédire les fonctions potentielles de ces derniers. Pour ce faire, des programmes de recherche d'homologie de séquences sont utilisés : lorsque la séquence d'acides aminés associée à un gène ressemble à une protéine connue, on considère qu'elle a une fonction homologue. Associer une fonction à un gène, c'est procéder à l'annotation fonctionnelle du gène.

Des algorithmes d'annotation qui recherchent des similarités (de séquence, motif, structure, etc.) pour prédire automatiquement la fonction d'un gène ont été développés. La recherche automatique peut ensuite être complétée par une annotation manuelle, le risque de la recherche automatique étant que si des erreurs d'annotation sont introduites dans les bases de données elles risquent d'être propagées et difficiles à tracer à long terme.

La démarche adoptée pour définir la fonction des gènes renvoie en grande partie à de l'alignement de séquences, point que nous avons abordé précédemment. Ce pan de la recherche biologique fournit un exemple d'application aux outils d'alignement de séquences (et ici dans une variante d'alignement de protéines) mais il n'y a pas pour moi plus à en dire que ce qui a été évoqué précédemment.
On pourra toutefois observer que les biologistes comparent les séquences de protéines en postulant qu'à séquences d'acides aminés proches, on peut associer des fonctions similaires. Existe-t-il de telles associations en transport ? Peut-on prétendre qu'à profils proches, des individus ont des comportements proches ? Ou encore à séquences d'activités proches, agendas proches (incluant les notions de durées, de mode de déplacement) ?

2.2.2.2. Génétique d'association : génotype et phénotype

Dans ce volet des sciences génétiques, le but est de rechercher directement des relations entre les caractéristiques phénotypiques et l'ADN des organismes. L'objectif est d'associer des

caractères exprimés à des gènes en se basant sur l'étude statistique des corrélations entre les variations génétiques et les variations des caractères.

La génétique d'association a de nombreux domaines d'applications. Elle sert en agronomie afin de produire par croisement des espèces végétales plus résistantes ou plus performantes. D'autre part, la génétique d'association sert en médecine dans le cadre de l'épidémiologie génétique qui se définie comme l'étude de facteurs génétiques qui interviennent dans le déterminisme d'une maladie. En effet, de nombreuses maladies sont liées en partie à des facteurs génétiques. Parmi ces maladies, on distingue les maladies monogéniques (on en répertorie environ 6000) qui ne sont dues qu'à un seul gène ; on parle de maladies mendéliennes. Parmi ces maladies on peut citer la mucoviscidose, la drépanocytose, l'albinisme, … D'autre part, il existe des maladies multifactorielles, dites complexes, dont les causes sont d'origines diverses et sont généralement la combinaison de facteurs génétiques et environnementaux. Dans ce cas les gènes impliqués sont plusieurs ; ceux-ci sont dits gènes de prédisposition ou gènes de susceptibilité. Parmi ce second type de maladies on trouve les cancers, le diabète, les maladies cardio-vasculaires, la schizophrénie, ...

Le principe de base des études d'association est de tester le degré d'association entre un trait donné et un gène ou plus fréquemment un SNP (Single Nucléotide Polymorphisme). Il s'agit en fait de déterminer s'il y a une corrélation ou indépendance entre le trait observé et le SNP. Dans ce cas, un simple test du Khi-deux permet de déterminer s'il y a corrélation ou non. Si l'on souhaite tester le trait observé pour plusieurs covariables (SPN, sexe, âge, environnement...) on peut recourir à des modèles statistiques (modèle de poisson, …) et effectuer des tests d'indépendance, ces tests permettent de quantifier le degré d'évidence d'association mais pas de fournir une 'mesure' de l'association. Une autre façon de procéder est d'utiliser un modèle de régression logistique afin de déterminer l'influence de chaque paramètre ; en outre ces modèles permettent de mettre en évidence une mesure d'association ("odds ratio"). Enfin, il est possible de tester des traits de caractère continus et d'utiliser un modèle de régression linéaire.

Lorsque l'on test un gène ou un ensemble gènes pressentis pour un trait, on se trouve dans une approche gènes-candidats. Cette approche permet de limiter la complexité du problème en réduisant le champ de recherche des facteurs. La contrepartie d'une telle approche est qu'elle ne permet pas d'appréhender de façon exhaustive les causes génétiques des maladies.

Une autre approche consiste à effectuer une étude d'association sur l'ensemble (ou du moins une grande partie) du génome sans a priori. On parle d'étude gemone-wide (GWA, genome-wide association). Ce type d'études a permis d'identifier de nombreux gènes responsables de maladies monogéniques, toutefois cette approche est plus difficile à mettre en place pour des maladies multifactorielles dont les gènes ont des influences et des effets plus modérés.

Pour les études d'association sur l'ensemble du génome il existe deux types d'approches : l'approche simple marqueurs et l'approche multi-marqueurs.
Un marqueur génétique est un gène ou une séquence polymorphe d'ADN aisément détectable grâce à un emplacement connu sur un chromosome (définition de l'encyclopédie Wikipédia).

Approche simple-marqueur

Dans les approches simples marqueurs, les marqueurs sont traités un à un. Cela revient à considérer que les marqueurs sont indépendants les uns des autres. On est alors dans le cas évoqué précédemment où l'on compare un trait vis-à-vis d'un marqueur. Les techniques employées sont des méthodes de tests statistiques, de régressions logistiques ou de modèles linéaires généralisés.

Dans le cadre des travaux en épidémiologie génétique ces méthodes sont appliquées à la recherche d'association marqueur-maladie. Les études d'association peuvent prendre en considération soit des génotypes (degré d'association de couples d'allèles) soit des allèles (degré d'association d'allèles).
Il existe également des travaux statistiques de combinaisons de tests qui visent à améliorer la finesse des prédictions.

Approche multi-marqueurs

Les maladies multifactorielles mettent en jeu différents gènes, or les gènes ne sont en réalité pas indépendants : d'une part il existe des liaisons alléliques mises en évidence par le déséquilibre des liaisons et d'autre part certains gènes peuvent "s'influencer'" (synergie ou opposition).

Dans l'approche multi-marqueurs, les marqueurs ne sont plus considérés comme indépendants les uns des autres. Les méthodes d'association multi-marqueurs reposent en outre sur la notion de déséquilibre des liaisons qui traduit le phénomène d'association préférentielle des allèles de loci différents : les marqueurs sont associés à leurs voisins par le déséquilibre des liaisons.

Citons les différents types de modèles développés :

- Régression Logistique, comme pour l'analyse simples marqueurs à la différence que les divers marqueurs sont des variables prédictives pour le marqueur à prédire.

- Approche méta-statistique par sommes de statistiques, cette approche combine l'information pertinente portée par un ensemble de marqueurs par l'intermédiaire de sommes de statistiques d'association simple-marqueurs.

- Approche combinatoire, le but ici est de passer par la recherche des combinaisons de génotypes les plus pertinentes pour expliquer la maladie

- Approche par partitionnements récursifs – utilisation d'arbres de discrimination et forêt aléatoire, partitionne de façon récursive l'ensemble des individus en sous-ensembles plus homogènes d'un point de vue de leur statut.

L'ensemble de ces méthodes sont répertoriées et décrites plus en détail dans la thèse de Mickaël Guedj, *Méthodes Statistiques pour l'Analyse des Données Génétiques d'Association à Grande Échelle*, 2007.

Mourad, Sinoquet et Leray (2010) s'intéressent à l'utilisation de modèles statistiques graphiques (Bayesian network, Markov random field, hidden Markov model, variable-length Markov model) pour les études génétiques d'association en prenant en compte les déséquilibres de liaison. Ils concluent en affirmant que ce type de modèles peut être utilisé pour une grande variété d'applications et notamment dans le cadre des études d'association gènes-candidat et genome-wide pour lesquels ce sont des outils puissants mais qui présentent néanmoins un certain nombre de limites ; ils évoquent notamment comme limite la diversité des modèles existants parmi lesquels il faut choisir le plus approprié.

Les méthodologies mises en œuvre dans le cadre de la génétique d'association et de l'épidémiologie génétique sont des méthodologies de recherche de relations (et de quantification) entre des causes (les gènes à la base de l'organisation et du fonctionnement des organismes) et des conséquences (les caractères ou les maladies et leurs symptômes).

Une analogie vient assez rapidement entre biologie et transport. En effet, il est possible de voir la séquence d'ADN comme le profil d'un voyageur et un caractère comme une 'propriété' de son agenda ou un type d'activité. Ainsi, les travaux des biologistes pourraient servir dans la recherche de prédisposition des individus à adopter tel ou tel comportement, pouvant servir par la suite à définir des règles de création d'agendas, ou à établir des critères afin de déterminer la vraisemblance des agendas.

Dans les faits, les outils les plus 'simples' directement issus du domaine de la Statistique tels que la règle du khi-deux, la régression logistique,... sont déjà utilisés par les chercheurs dans le domaine du transport pour l'élaboration de modèles économétriques ou l'élaboration des fonctions d'utilité.
Pour ce qui est des outils et techniques élaborés pour la recherche de traits complexes qui mettent en œuvre des facteurs multiples, ceux-ci semblent présenter un intérêt notoire. En effet, dans la mesure où chaque activité peut être vue comme "un symptôme" lié à un certain nombre de caractéristiques d'un individu ainsi qu'à des facteurs environnementaux (structure du ménage, jour de la semaine, ...), les recherches et les résultats novateurs des biologistes en épidémiologie médicale peuvent constituer une ressource exploitable.

2.2.3. Des outils statistiques "puissants"

On constate dans les divers travaux des biologistes et notamment dans le domaine bioinformatique l'utilisation récurrente de modèles de chaînes de Markov et de modèles de Markov cachés.

Cela peut s'expliquer par la structure des données qui se présentent sous forme de chaînes de caractères d'alphabets de taille restreinte (4 lettres pour l'ADN et l'ARN et 22 lettres pour les protéines). De plus ces 'textes' forment un ensemble codant qui répond à des contraintes et suit certaines règles que les chercheurs tentent de mettre à jour. Ainsi, puisque les séquences étudiées ne sont pas le seul fruit du hasard, il est possible d'en faire une analyse statistique et d'adopter des approches probabilistes.

Les premiers à adopter une approche de modélisation probabiliste sont David Haussler et al. qui présentent en 1992 leurs travaux sur l'utilisation de chaînes de Markov pour effectuer des alignements multiples de séquences de protéines. Ce type d'approche a rapidement été adopté par les biologistes qui ont utilisé entre autres des Modèles de Markov Cachés pour dans de nombreux travaux en alignement de séquences mais aussi en recherche de gènes, pour la classification de protéines, en phylogénie, etc.

Les chercheurs en génétique ont vu dans le modèle de Markov caché un outil mathématique adapté à l'analyse et au traitement de l'information contenue dans les séquences biologiques.

Si on se place dans le cadre de la modélisation d'agendas, un agenda peut être modélisé et codé de différentes façons mais une façon des plus simples de représenter une séquence d'activités est de l'écrire sous la forme d'une chaine de caractères chaque caractère représentant un type d'activité. De plus, comme les séquences d'ADN, les séquences d'activités répondent à certaines règles d'organisation (cf. travaux en sociologie de Chapin et Hägerstrand) ainsi les séquences d'activités ne constituent pas des ensembles définis de façon complétement aléatoire. Par conséquent, l'utilisation de chaîne de Markov pour analyser et générer des agendas (ou du moins des séquences d'activités) semble fortement envisageable.

2.2.3.1. Modélisation des séquences d'ADN par des chaînes de Markov – Modélisation de séquences d'activités par des chaînes de Markov

Ce chapitre est rédigé sur base d'un cours de l'école polytechnique disponible sur internet à l'adresse : http://biology.polytechnique.fr/biocomputing/data/Statseq.pdf (auteur inconnu).

On retrouve dans la littérature des travaux de modélisation de séquences d'ADN par des chaînes de Markov. Par exemple le cours cité plus haut présente une méthodologie pour modéliser l'ADN de dinosaure. Le principe est de générer une séquence aléatoire par chaînes de Markov à partir de fréquences observées chez d'autres espèces animales « proches » (ex. table de données pour les vertébrés).

Il est possible de prendre en compte les successions de symboles (lettres codant pour les nucléotides). Les fréquences de 'successions' entre symboles qui permettent de déterminer les probabilités de transition. La probabilité d'un symbole dépend du symbole précédent. Un processus de Markov permet alors de générer une séquence de nucléotides.

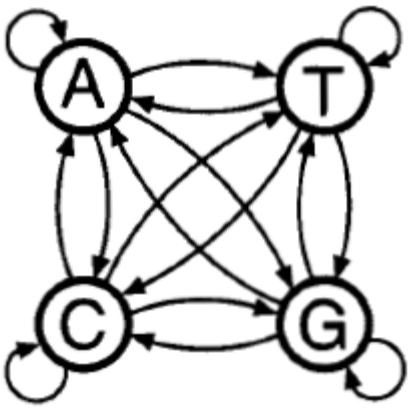


Figure 20 Modèle de Markov pour l'ADN (Durbin, Eddy, Krogh, et Mitchison, 1998, p.48)

De la même façon il est possible de définir des probabilités de transition entre les types activités et construire des séquences d'activités. Plusieurs approches peuvent être envisagées, il est possible de construire des séquences d'activités avec un changement d'état à chaque étape du processus de Markov, ou il est aussi possible d'envisager que chaque état correspond à une plage horaire (de 5 minutes par exemple), ou on peut aussi chercher à appliquer un modèle de Markov continu.

Par ailleurs, il serait possible de complexifier le modèle en définissant des probabilités qui varient en fonction des plages horaires avec par exemple de plus fortes probabilités de transition vers une activité « Restauration » aux heures de repas ».

2.2.3.2. Autres pistes de réflexion

Dans ce chapitre, nous allons passer en revue quelques grands outils qui permettent de travailler avec des séquences de données, d'informations ou de variables.

Le but est de présenter brièvement quelques outils issus des domaines de la statistique et de la probabilité, d'en donner des applications connues et d'établir un lien avec la recherche en modélisation de la demande de transport.

Chaînes de Markov

Une chaîne de Markov est un ensemble de variables aléatoires (X_n) dont l'état de la variable X_n dépend uniquement de la variable X_{n-1}.

Elles sont utilisées dans des modèles de diffusion, de marche aléatoire, de ruine du joueur, … En biologie (bioinformatique), elles sont utilisées pour modéliser les relations entre les symboles successifs d'une séquence d'ADN (Durbin, Eddy, Krogh, et Mitchison, 1998, p.48). C'est ce que nous avons évoqué dans le paragraphe précédent.

Les chaînes de Markov font intervenir des « états observables » et utilisent des probabilités connues ; ainsi, ils permettent généralement de générer des séquences. Nous avons évoqué précédemment la possibilité d'utiliser des modèles de Markov pour créer des séquences d'activités.

Modèles de Markov Caché

Dans un modèle de Markov caché les états des variables ne sont pas « visibles », mais chaque état génère un état observable de paramètres « visibles » (ou mesurables). La séquence de paramètres visibles (v_m) est aussi une chaîne de Markov.
Les modèles de Markov cachés permettent d'énumérer toutes les séquences possibles d'états cachés qui peuvent permettre d'observer un état observable donné. Il est de plus possible de définir la probabilité pour chaque séquence cachée de donner la séquence observée.

Les modèles de Markov sont utilisés dans de nombreux domaines. Ils trouvent des applications directes dans des problèmes d'évaluation, de décodage, de messages cryptés et dans les méthodes d'apprentissage. Ainsi, ils sont notamment utilisés en reconnaissance de formes, en intelligence artificielle et en traitement du langage.

Ils sont également très présents en biologie (bioinformatique) où ils sont utilisés en alignement de séquences, pour la découverte de gènes, en génétique d'association et en phylogénie pour la construction d'arbres.

Nous avons déjà abordé la plupart de ces thèmes précédemment : en transports les modèles de Markov cachés pourront notamment servir pour la classification et l'analyse des séquences d'activités et pour étudier les interactions entre les caractèristiques des individus et leurs agendas.

Méthode Monte Carlo par chaînes de Markov

La méthode de Monte Carlo par chaînes de Markov est une méthode d'échantillonnage à partir d'une distribution de probabilité. Le but est de générer aléatoirement un ensemble de vecteurs suivant une distribution de probabilité.

Ismaïl Saadi (2016a), utilise une méthode de Monte Carlo par chaînes de Markov pour créer une population synthétique d'agents pour un modèle de transport et développe un autre modèle (2016b) qui emploie un modèle de Markov étendu pour créer une population synthétique d'agents.

A la lumière de ces travaux, il est possible de mettre en évidence le fait que non seulement les agendas ont une structure de séquence mais aussi les « phénotypes » des individus (ou des agents). Ainsi, des outils de biologie peuvent servir à l'étude et à la création d'agendas mais certains outils utilisés par les biologistes peuvent s'appliquer à d'autres types de séquences comme par exemple aux séquences de caractéristiques des individus.

Nous avons évoqué ici certains modèles statistiques, d'autres modèles encore peuvent être considérés notamment tous les modèles dérivant du modèle de chaînes de Markov mais aussi les méthodes d'inférence bayésienne, les réseaux de neurones, réseaux bayésiens dynamiques, etc... En outre, certains de ces modèles et d'autres encore peuvent être (re)trouvés dans les travaux des biologistes.

2.2.4. Création de nouvelles séquences.

2.2.4.1. Drosophila Synthetic Population Ressource (DSRP)

Afin de mener une étude détaillée de l'ADN et de décrire les nucléotides responsables de caractères phénotypiques complexes, en 2011 Elizabeth G. King et al. créent la Drosophila Synthetic Population Ressource (DSPR). Différents travaux théoriques de recherches sur l'ADN sont menés sur des drosophiles (*Drosophila melanogaster*), petites mouches, qui présentent l'avantage d'avoir un cycle biologique court (de l'ordre de 12 jours, ce qui permet d'obtenir 25 générations par an) et un génome réparti sur quatre paires de chromosomes et complètement séquencé depuis le début des années 2000.

La DSPR est un panel de 1700 lignées pures recombinantes (RILs), résultats du croisement de 15 lignées pures fondatrices. Les séquences ADN des lignées fondatrices sont disponibles ainsi qu'une carte génétique détaillée pour chaque RIL.

King et al. (2011) explique la méthode suivie. La première étape de la démarche consistait à produire les RILs en croisant différents groupes de drosophiles sur plusieurs générations.

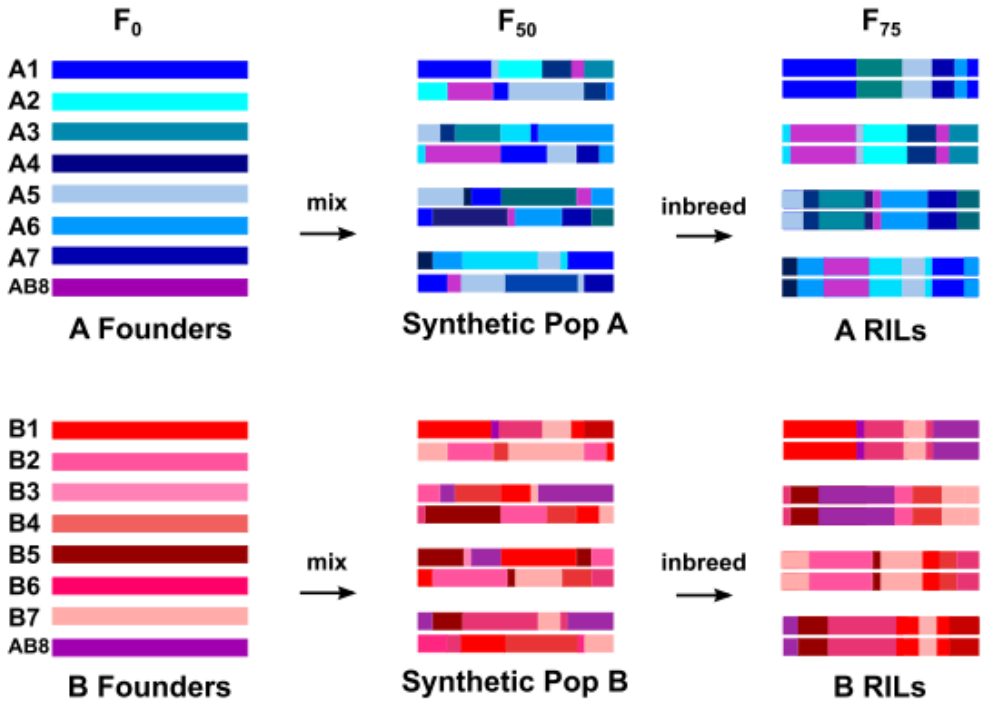


Figure 21 Schéma de reproduction suivi afin de créer les RILs (source : E. G. King et al., 2011 p.1559)

Le processus de croisement démarre avec 8 groupes d'individus possédant un ADN quasiment identique au sein de chaque groupe. Ces différents groupes sont croisés (ex. A1xA2, A2xA3,..., AB8xA1), on obtient ainsi une première génération F_1 de mouches filles. Ces dernières sont utilisées pour produire une génération F_2. Ensuite à partir de la troisième génération les mouches sont reparties en deux groupes isolés l'un de l'autre et tous les 12 jours

(durée moyenne d'un cycle de reproduction) les adultes sont retirés des deux environnements de culture.

Ce brassage génétique s'effectue sur 50 générations puis pour chaque sous-population (2x2), des individus mâles et femelles sont sélectionnés pour former 576 paires. Pour chaque génération suivante un à trois mâle et une femelle sont sélectionnés pour donner naissance à la génération fille. Après 25 générations, on peut considérer que la lignée obtenue est une "lignée pure".
Dans sa démarche E. G. King et al. procèdent ensuite à un séquençage de l'ADN des 1700 RILs puis ils construisent un modèle de Markov Caché (HMM) qui permet de déterminer l'ancêtre à l'origine de chaque segment d'ADN dans chaque RIL.

Dans le domaine de la génétique, et en particulier pour la communauté de chercheurs qui travaille sur les drosophiles, la DSPR est une ressource clé et une étape majeures dans l'avancé de leurs travaux. En outre la DSPR présente de nombreuses caractéristiques qui la rendent très avantageuse à utiliser : grande diversité des génomes des fondateurs, obtention de RILs en fin d'expérience, cartographie génomique de bonne résolution,...

Mais quel intérêt et quel rapport avec notre problématique de transport me direz-vous ?
En fait, ce qui est marquant dans ce travail sur les drosophiles c'est de voir comment à partir de seulement 2x7+1 soit 15 individus distincts on obtient après 25 générations seulement près de 1600 individus distincts. En outre, créer de la diversité dans une population peut présenter un intérêt lors de la création de populations synthétiques au lieu de faire de simples copies d'individus observés.

Par ailleurs, les mécanismes de croisement et de sélection naturelle mis en jeu lors de la création de nouvelles générations d'individus sont à la fois simples et extrêmement puissants ; il suffit de regarder la diversité et la complexité du vivant pour s'en rendre compte. C'est cette efficacité qui a mené les chercheurs à reprendre ces mécanismes dans d'autres domaines de recherche. En outre, on trouve dans la littérature des modèles d'optimisation qui s'inspirent du processus de sélection naturelle. En outre, Charypar et Nagel (2004) proposent un modèle pour générer un agenda journalier avec un algorithme génétique.

2.2.4.2. Les algorithmes génétiques

Les algorithmes génétiques sont des modèles computationnels inspirés de la théorie de Darwin sur l'évolution des espèces. Cette dernière met en lumière trois principes majeurs : le principe d'hérédité, le principe de variation et le principe d'adaptation.

Le principe d'hérédité renvoie à l'hérédité des caractères d'une génération à la suivante ; les descendants ont un phénotype semblable à celui de leurs ancêtres. Dans la nature cela permet la conservation des caractères (et notamment des meilleurs caractères pour l'espèce).

Le second principe, principe de variation, illustre l'unicité de chaque individu dans une population. Chaque nouvel individu est à la fois semblable (cf. premier principe) et différent de ses parents. De plus, chez certains individus de nouveaux caractères inédits peuvent apparaître, on parle alors de mutation génétique.

Enfin, le principe d'adaptation fait référence au fait que, sur une période de temps relativement vaste, les espèces ont tendance à "s'adapter" à leur milieu. C'est en fait le mécanisme de la sélection naturelle : les individus avec des caractéristiques qui les rendent plus performant dans leur habitat (ou plus séduisants) auront de meilleures capacité de survie et de reproduction.

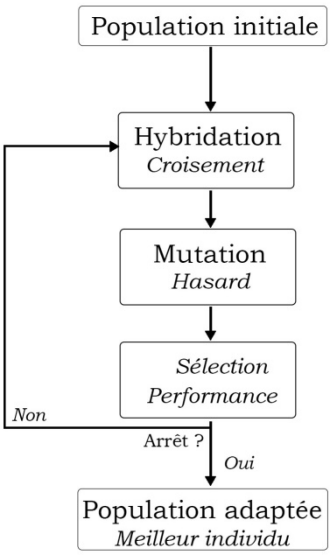


Figure 22 Schéma de principe des algorithmes génétiques

Ces trois principes peuvent être assimilés à trois opérations dans un algorithme : la reproduction (hybridation, croisement des caractères), la mutation (altération aléatoire des caractères) et la sélection (choix des meilleurs individus).
Les premiers travaux sur des algorithmes génétiques datent des années 1960. Les scientifiques de domaines variés étaient intéressés par l'efficacité des mécanismes relativement simples de l'évolution qui ont permis de créer toute la complexité du vivant telle qu'on la connait aujourd'hui. C'est J.H. Holland qui introduit le terme d'algorithme génétique et qui fonde les bases théoriques de ce domaine de recherche lorsqu'il publie en 1975 les premiers résultats de ses travaux dans son ouvrage "*Adaptation in Natural and Artificial System*". Pour J.H. Holland, c'est la notion d'adaptation qui est centrale et les champs d'application sont très nombreux ; c'est son étudiant K.A. De Jeong (1992) qui focalisera son travail sur l'utilisation des algorithmes génétiques pour résoudre des problèmes d'optimisation.

Et, il est vrai qu'aujourd'hui, les algorithmes génétiques sont principalement utilisés pour trouver des solutions à des problèmes d'optimisation.

Les entités du domaine de la génétique sont ainsi transposées dans un problématique de recherche d'une solution optimale ou à défaut la moins mauvaise. Ainsi, la population est un ensemble (restreint) de solutions potentielles, l'individu est une solution, un chromosome (séquence ADN) décrit l'individu (reprends ses attributs) et enfin, un gène est un attribut, une caractéristique de l'individu.

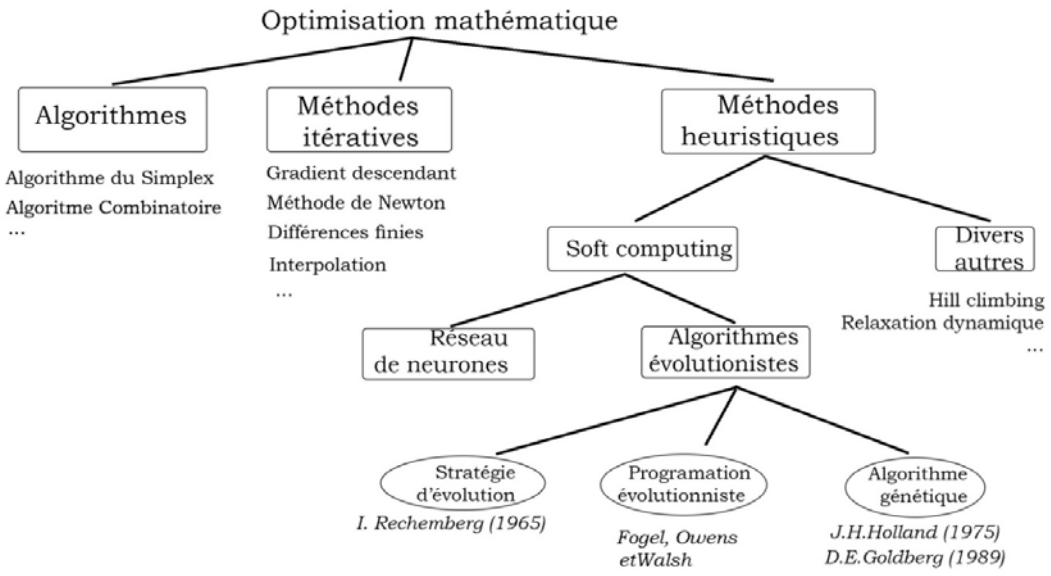


Figure 23 Les algorithmes génétiques parmi les modèles d'optimisation

L'algorithme consiste en trois étapes récursives.

- Etape d'hybridation (ou reproduction)

Cette étape renvoie au principe d'hérédité : les "chromosomes" d'une paire d'individus dits parents sont croisés pour produire un nouvel individu. La méthode généralement utilisée à cette étape est le croisement, ou enjambement ou encore crossing-over ; on peut réaliser un simple enjambement ou un double enjambement voire un enjambement multiple.

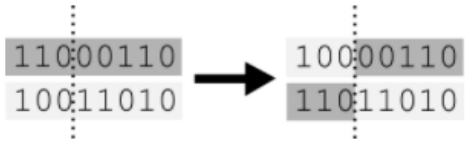


Figure 24 Principe du simple enjambement

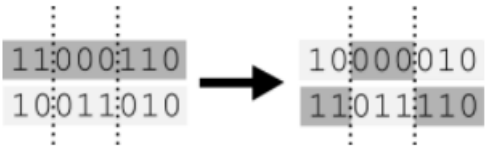


Figure 25 Principe du double enjambement

P. Deng montre qu'utiliser un plus grand nombre d'enjambements pour générer plus de chromosomes "fils" permet d'augmenter la vitesse de convergence.

- Etape de mutation

La mutation consiste à modifier la valeur d'un ou plusieurs gènes d'un chromosome. Ce mécanisme permet de mettre en application le principe de variation ; via cette action, de nouveaux caractères peuvent être intégrés à une population et des individus inédits peuvent être créés. Dans les algorithmes génétiques c'est étape vise en outre à éviter que l'algorithme ne stagne sur un extrema local.

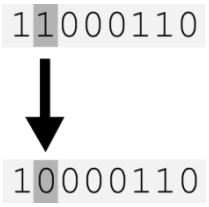


Figure 26 Principe de la mutation

- Etape de sélection

La sélection traduit le principe d'adaptabilité. Cette étape consiste à sélectionner les meilleurs individus de sorte à garder une population de taille constante. L'élément central de l'opération de sélection est la fonction de fitness ou fonction d'adaptation. Elle permet de déterminer pour chaque individu un "score d'adaptation" qui traduit un niveau de performance.

Il existe plusieurs techniques possibles de sélection :

- la sélection par rang : les individus avec le meilleur score d'adaptation sont sélectionnés.
- la sélection par roulette : les individus ont une probabilité d'être choisie proportionnelle à leur score d'adaptation (cette méthode peut être effectuée avec ou sans remise).
- la sélection par roulette sur les individus restants : dans ce cas pour chaque individu on calcul la valeur $N \times f_i / \sum_i f_i$ (ou N est le nombre d'individus à sélectionner). La partie entière de la valeur calculée correspond à un nombre d'individus sélectionnés pour engendrer la nouvelle génération ; la partie décimale correspond à la probabilité de sélectionner une copie supplémentaire de cet individu. On effectue une sélection par

roulette pour déterminer les derniers individus et obtenir une population de N individus. (version avec remise)

- la sélection par tournois : Là encore on peut procéder avec ou sans remise. Une variante consiste à sélectionner deux individus au hasard, à "choisir" un gagnant en considérant pour chacun des deux individus une probabilité d'être choisi proportionnelle à son score d'adaptation. Le "gagnant" est sélectionné (et n'est pas remis dans le tirage), le perdant est remis dans la liste des prétendants à la sélection. On réitère l'opération jusqu'à obtenir le nombre d'individus souhaité.
- la sélection uniforme : la sélection se fait aléatoirement sans tenir compte de la valeur d'adaptation. (Ne permet pas d'atteindre un optimal).
- Sélection par élitisme : une partie des individus est sélectionnée par rang puis une autre méthode de sélection est utilisée pour sélectionner les individus restants.

2.2.4.3. Application des algorithmes génétique au domaine des transports

Les algorithmes génétiques existent sous une forme canonique que nous venons de décrire et peuvent être utilisés dans une multitude de domaines d'application. Ce sont Charypar et Nagel (2003) qui dans les années 2000 créent un modèle basé sur l'activité qui s'appuie sur la structure des algorithmes génétiques. Leur but est d'utiliser un algorithme génétique pour générer des agendas journaliers et d'en évaluer l'efficacité de calcul.

Conscients que la création d'agendas est un problème combinatoire très vaste dont il est impossible d'énumérer toutes les possibilités, Charypar et Nagel cherchent dans leur démarche une solution "suffisamment bonne" au problème et non pas forcément la meilleure. En outre, ils constatent que bien souvent les agendas réels sont loin d'être optimaux. Dans cette optique les algorithmes génétiques semblent être un outil de choix qui doit au moins être essayé ; ce sont Abraham et Hunt (2002) qui en propose l'utilisation dans le domaine des aménagements de transport.

Appliqué à notre problème, la population de solution est un ensemble d'agendas envisageables pour un voyageur. Pour la fonction d'adaptation une fonction d'utilité adaptée est construite. La population de départ est créée de façon aléatoire, puis les descendants sont créés par croisement et mutation. La méthode de sélection utilisée est une méthode par rang : les meilleurs agendas sont conservés, les moins bons sont écartés afin de maintenir une population de taille constante.

Enfin, un dernier point crucial est la question du codage d'un chromosome c'est à dire ici d'un agenda. Chaque agenda est représenté par quatre tableaux :

- un vecteur (tableau à une dimension) de bits enregistre les activités menées dans la journée ; pour un ensemble défini d'activité, on détermine si une activité est réalisée (1) ou non (0).
- un second vecteur détermine l'ordre de réalisation des activités. Il s'agit d'un tableau de permutation qui détermine l'ordre de l'ensemble des activités du set initial ; il ne tient pas compte du fait que les activités soient effectivement réalisées ou non.
- Un troisième tableau lie chaque activité à sa localisation (pour l'individu).
- Un dernier tableau associe à chaque activité une durée ; ce tableau contient également le point de départ de la journée.

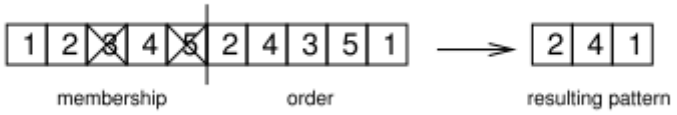


Figure 27 Encodage du schéma d'activité (tableau 1 et 2) (source : Charypar et al. (2008), p.21)

Les opérations d'hybridation (croisement) et de mutation opèrent sur les deux premiers tableaux (participation et ordre) ainsi que sur le quatrième (durées) mais le tableau des localisations ne sera jamais modifié.

La fonction d'adaptation s'inspire des fonctions d'utilité, la question étant de savoir qu'est-ce qu'un "bon" agenda. Charypar et Nagel décident d'utiliser une fonction qui somme les utilités de chaque activité réalisée à laquelle il ajoute la somme des utilités (négatives) liées aux trajets.

$$F = \sum_i U_{act,i}\,(type_i, start_i, dur_i) + \sum_i U_{travel,i}\,(loc_{i-1}, loc_i)$$

$type_i$: type d'activité i
$start_i$: date de début de l'activité i
dur_i : durée de l'activité i
loc_{i-1} : localisation de l'activité i-1
loc_i : localisation de l'activité i

avec $U_{act,i} = U_{duration,i} + U_{wait,i} + U_{late.arrival,i} + U_{early.departure,i} + U_{Short.duration,i}$

A l'exception de l'utilité relative à la durée de l'activité, tous les termes varient linéairement. Les auteurs soulignent que la fonction d'adaptation est un élément clé des algorithmes génétiques mais que néanmoins leur proposition est seulement une proposition et qu'elle peut être facilement revue ou remplacée sans remettre forcément en cause tout l'algorithme. Pour plus de détail sur le calcul de l'utilité je vous invite à lire Charypar, D. et al. (2008) Efficient algorithms for the microsimulation of travel behavior in very large scenarios (doctoral thesis), p.25-28.

Une fois leur modèle construit, Charypar et Nagel l'ont mis en application sur quelques cas "simples". L'algorithme a fourni des résultats satisfaisants. Par ailleurs, ils affirment que l'algorithme génétique ne dépends pas essentiellement de la fonction d'adaptation utilisée ; il permettra quelle que soit la fonction utilisée, de faire progresser par itérations une solution vers une meilleure solution et permettra donc toujours d'obtenir une solution satisfaisante (au sens de plausible).

Enfin, Charypar et Nagel mettent en avant le caractère très flexible des algorithmes génétiques qui offrent un cadre de travail relativement souple. Cette flexibilité permet d'intégrer assez facilement plus d'aspects encore qui entrent en compte dans la prise de décision que ce qu'ils ne l'ont fait eux-mêmes. Par exemple les interactions au sein du foyer pourraient être intégrées ou même encore l'aspect de l'accès limité à l'information de la part des voyageurs pourrait être modélisé (à part) et intégré dans le modèle. Mais, en augmentant la complexité du modèle, on augmente aussi la complexité informatique (temps de calcul). Les auteurs proposent d'envisager de coupler l'utilisation d'un algorithme génétique avec un processus séquentiel de planification des activités comme celui décrit par Doherty and Axhausen (1998) et de l'appliquer dans le modèle ALBATROSS (Arentze et al., 2000).

En 2005, K. Meister et al. s'appuient sur les travaux de Charypar et Nagel pour élaborer un algorithme génétique qui permet de modéliser des agendas pour les membres de ménages et plus seulement pour des individus isolés.

Afin d'attribuer les activités aux individus, toutes les activités possibles sont répertoriées dans le modèle. Chaque activité est associée à une liste de caractéristiques propres. On précise notamment si l'activité a un caractère obligatoire ou non, les heures de réalisation (heures d'ouverture s'il y a lieu), si elle est réalisée individuellement ou en groupe...

Encore une fois la fonction d'adaptabilité utilisée est une fonction d'utilité. Celle utilisée par K. Meister et al. est la somme des fonctions d'utilité de chaque membre du ménage. Les interactions entre membres d'un même foyer sont prises en compte via la fonction d'adaptabilité. Le but étant de traduire d'une part le partage de certaines tâches et d'autre part les désirs de réaliser des actions avec des proches. Par ailleurs, le modèle permet de prendre en compte le partage des ressources de transport disponibles pour le foyer.

Pour le codage de leurs agendas K. Meister et al. se basent sur le modèle proposé par Charypar et Nagel et reprennent leurs quatre tables (scheduled (binary), sequence (integer), location (integer), time allocation (real double)) auxquelles ils joignent une cinquième table pour le mode de transport (mode (integer)).

Pour l'initialisation de l'algorithme, les populations d'agendas sont générées aléatoirement. K. Meister et al. définissent 50 classes d’agendas ; une population est composée d'un et un seul agenda pour chacune des classes. Lors de la réalisation de l'algorithme chaque agenda créé est comparé (valeur de la fonction d'utilité) avec l'agenda qui corresponds à la même classe et le meilleur agenda est sélectionné. En procédant ainsi, l'objectif est de décrire au mieux l'ensemble de solutions afin de converger vers un optimum global et de ne pas rester bloquer sur un optimum local.

L'algorithme a été testé pour une famille de trois personnes. Il permet de trouver une bonne solution pour le problème considéré à savoir établir un agenda cohérent pour l'ensemble des individus du ménage est qui maximise également l'utilité pour l'ensemble du ménage. Toutefois, cet algorithme présente un problème de complexité, le temps de calcul est trop élevé. Par ailleurs, la fonction d'utilité doit encore être calibrée.

2.3. Conclusion et travaux futurs

Comme le domaine de la gestion de la demande de transport, le domaine de la biologie est un domaine d'étude à la fois vaste, complexe et pointu. On trouve en outre dans la littérature une multitude de travaux allant de l'alignement de séquences à la modélisation des structures secondaires des protéines en passant par les études de fonctions des entités du vivant, la génétique de l'évolution, etc. De plus dans chacun de ces sous-domaines d'études, il existe une variété d'alternatives méthodologiques. De plus, selon les applications visées, les approches peuvent différer pour une même problématique. Par exemple dans le cadre de la prédiction de gènes, certains chercheront à analyser tout le génome, d'autres chercheront les régions codantes spécifiques pour la synthèse de protéines particulières, d'autres encore procéderont en recherchant des exons spécifiques. La démarche utilisée dépend en grande partie du domaine d'application des chercheurs (médecine, biotechnologie, agronomie,...) et de l'utilisation visée.

Les voies à explorer sont ainsi très nombreuses.

Dans ce second chapitre, nous avons mis en avant certains travaux de recherche menés par les biologistes. Nous avons abordé certaines possibilités d'appropriation de techniques et de méthodes développées dans les domaines de la biostatistique et de la bioinformatique transposables à des modèles ou à des sous-modèles de transport.
Nous avons notamment commencé à mettre en avant certaines similitudes entre la biologie et la modélisation de phénomènes de transport basée sur les activités. En effet, dans les deux cas, les objets manipulés sont des informations organisées en séquences, caractérisant une multitude d'individus distincts.

Dans des travaux futurs, l’objectif serait d’approfondir et de développer les pistes évoquées dans ce chapitre afin de déterminer plus en détail les utilisations possibles des outils et des techniques employés par les biologistes afin de les appliquer au transport. De cette démarche de recherche d’alternatives aux modèles existants et de recherche des pistes d'amélioration pourraient naître de nouvelles générations de modèles plus performants.

D’autres pans du domaine de la biologie génétique, des biostatistique et de la bioinformatique n’ayant pas été explorés pourraient être exploités et prometteurs. On pourrait ainsi s'intéresser par exemple aux travaux sur la construction d'arbres phylogénétiques et à la "transmission" de gènes entre les espèces à travers l'histoire de l'évolution.
Il serait également intéressant de pouvoir recroiser des travaux issus de réflexions d’ordre plus "méta". Je pense notamment à des publications décrivant dans le domaine du médical la problématique du manque de banques de données désagrégées. En effet, comme les chercheurs en transport, les biologistes sont confrontés au problème du "respect de la vie privée" et de l'anonymat des données qui forment un obstacle à leur travail. Le projet dataSHIELD (Jones et al., 2012) aborde cette problématique particulière et propose une solution afin de protéger l'identité des individus tout en conservant des données désagrégées.

Une autre problématique peut être soulevée sur la question de la gestion et de l’utilisation des données. En effet, il est possible de questionner l'utilisation exclusive de données d'enquêtes provenant du secteur d'étude pour générer une population synthétique, puis et c'est une autre problématique, pour générer des agendas. La question posée est celle de l'"universalité" des profils rencontrés ainsi que de l'"universalité" des comportements humains. Dans quelle mesure l'environnement et la culture influencent les comportements et les profils individuels et familiaux ? N'y aurait-il pas un intérêt à développer des bases de données communes plus "universelles» ?

De plus, au jour d'aujourd'hui, les sources de données utilisées pour la modélisation de phénomènes de transport sont différentes pour chacun des chercheurs et dépendent des pays et des lieux d'études. Il est concevable que certaines données recueillies dans un cadre non spécifique au transport telles que les données de recensement puissent être différentes d'un pays à l'autre. Toutefois, pour des données spécifiquement collectées pour l'étude et la modélisation des phénomènes de transport, il pourrait être envisageable que les chercheurs puissent établir une liste de caractéristiques à recueillir auprès des populations. Cela pourrait alors éventuellement permettre d'élaborer des banques de données communes telles qu'il en existe en biologie. Cela permettrait également de concevoir des modèles plus largement applicables. Aujourd'hui les modèles sont fortement dépendants de la disponibilité des données, est presque chaque grande ville développe son propre modèle.
Enfin, nous évoqueront les travaux de Sillanpää (2011) qui présentent des techniques pour tenir compte des phénomènes de stratification de population (origine inconnue des individus de plusieurs populations source) et de parenté cryptique (covariances inconnues entre individus à cause d'une parenté) qui causent une augmentation du nombre de faux positifs dans les études d'association. Ces techniques pourraient amener des éléments de réflexions dans l'hypothèse de travaux exploitant des bases de données ne se limitant plus seulement aux secteurs d'études.

Les domaines de recherche sont donc divers et variés, les approches et les analogies peuvent elles aussi être multiples. On pourra notamment évoquer les diverses possibilités de représentation des agendas. Un agenda peut se représenter comme une séquence d'activités symbolisées par des lettres distinctes dépendant de l'activité menée (Wilson, 1998). Il est également possible d'introduire une notion de durée et de représenter un agenda sous la forme d'une séquence de lettres qui définissent chacune le type d'activité mené pour un intervalle de temps (Shoval et Isaason, 2007). D'autres représentations encore sont possibles ; une séquence d'activités peut être représentée par deux vecteurs l'un déterminant si une activité est réalisée ou non et l'autre définissant l'ordre de réalisation des activités (Charypar et Nagel, 2003). Du type de représentation dépendront les outils et les techniques de création et d'analyse des agendas.
Par ailleurs, on peut aussi rechercher des méthodes pour créer des populations synthétiques ou pour analyser, classifier, ou générer des profils d'individus...

Dans le chapitre suivant, le but sera de développer un modèle qui approfondit une piste de recherche évoquée dans ce chapitre.

Chapitre 3

UN EXEMPLE D'APPLICATION : GENERER DES AGENDAS EN S'INSPIRANT DE MECANISMES BIOLOGIQUES A TRAVERS UN RAISONNEMENT PAR ANALOGIES

Lorsqu'au fil de mes recherches j'ai découvert les travaux d'E. King sur les populations de drosophiles, j'y ai immédiatement trouvé un intérêt (et une curiosité) pour le sujet qui nous occupe. En effet, avant même de découvrir les travaux de Charypar et Nagel, découvrir comment à partir d'un ensemble d'individus il était possible d'en "créer" de nouveaux qui sont à la fois différents de leurs parents et pourtant ressemblants.

Ma première idée était de créer un agenda fils à partir de deux "agendas parents" en sélectionnant des parents de profils proches de celui de l'individu pour lequel nous voulons créer un agenda. Le postulat au centre de ma réflexion est le suivant : deux individus qui présentent les mêmes caractéristiques (âge, sexe, emploi, diplôme, type de ménages, lieu d'habitation,...) seront fortement susceptibles d'avoir également des agendas semblables. Ainsi, cela revient à affirmer l'existence de corrélations entre les "attributs" d'un individu et son agenda de même qu'en génétique il existe un lien étroit entre les caractéristiques phénotypiques d'un individu et son ADN.

D'autre part, comme je l'ai déjà évoqué précédemment, les travaux d'E. King présentent un 'modèle' dans lequel, 8 individus donnent naissance à une diversité de 1600 individus. Or, un des principaux écueils des méthodes de création d'une population synthétique est de ne pas étendre la diversité des individus, seuls les types observés sont repris dans les modèles ; un type non observé reste absent du modèle. En créant de nouveaux profils en procédant par croisement et mutation il est possible d'augmenter la variabilité et d'interpoler les valeurs observées (voire d'extrapoler par le biais des mutations) pour créer une population d'individus plus continue et moins discrétisée.

Les travaux qui exploitent, les algorithmes génétiques ne vont selon moi pas assez loin dans leur démarche, ils utilisent un outil relativement flexible utilisé dans une grande variété de domaines. Si l'outil s'inspire de la biologie, ses utilisateurs en transport l'emploient sans chercher à établir d'avantage de liens entre génétique et transport ; c'est pour eux un outil d'optimisation suffisamment flexible pour être applicable au problème de création d'agenda et qui leur permet de trouver une solution approchée à un problème complexe d'optimisation. Or, d'après moi, il est possible d'aller plus loin. En effet, ce qui fonde ici mon travail est l'observation que certaines structures sont semblables entre le monde de la biologie (génétique) et du transport. Mon but dans ce troisième et dernier chapitre sera de mener une démarche expérimentale de recherche afin de proposer un nouveau modèle pour la création d'agendas en m'inspirant des mécanismes biologiques mis en avant à travers les travaux de E. King en établissement préalablement un cadre de travail par analogie entre domaines.

Le cas que je choisis de traiter sort un peu de la problématique fixée initialement dans la mesure où je ne vais pas, à proprement parler, utiliser un modèle statistique pour l'appliquer directement à un problème de planification des transports. Néanmoins, à travers ma démarche je vais tenter de mettre en avant un raisonnement par analogie entre le domaine de la biologie

et plus particulièrement de la génétique et celui de la modélisation des phénomènes de transport basés sur l'activité. Le but étant de montrer les passerelles, les liens et les analogies que l'on peut établir afin d'exploiter des outils de généticiens dans la modélisation en transport.

3.1. Raisonnement par analogie - mise en place des modèles

L'analogie entre transport et génétique n'est pas si évidente, ce n'est aussi flagrant que l'analogie entre hydraulique et électricité par exemple. Néanmoins il n'est pas absurde de voir certaines correspondances entre ces deux domaines. En effet, en génétiques, les chercheurs travaillent avec des individus décrits par des caractères phénotypiques d'une part et un génome d'autre part. L'ensemble des individus forment une population, leurs caractéristiques sont des attributs "visibles" et le génome est une séquence de nucléotides, regroupés en gènes qui déterminent ces attributs visibles. L'un des objectifs des généticiens est de cartographier les génomes des diverses espèces et de mettre en évidences les corrélations les caractères exprimés et les gènes qui en sont responsables. En transport, dans le cadre de la modélisation basée sur les activités, les activités sont une séquence à partir de laquelle sont déterminées les 'caractéristiques' de déplacement des individus d'une population humaine. Il est possible de faire un autre parallèle : les attributs (caractéristiques accessibles par le recensement ou par enquête) forment un phénotype plus ou moins fortement corrélé avec les séquences d'activités des individus et, comme les généticiens le font pour l'ADN, l'enjeu pour les chercheurs en transport est de déterminer comment sont corrélés activités et caractères.

En fait, la façon d'aborder le problème n'est pas unique et elle ne doit pas chercher à l'être. Ce qu'il est important de mettre en évidence c'est la ressemblance entre les 'objets' manipulés dans ces deux domaines qui ont des individus pour unité de travail auxquels sont associés de multiples séquences (séquences de caractères, séquences de gènes, séquences de protéines, séquences d'activités,...) corrélées entre elles.

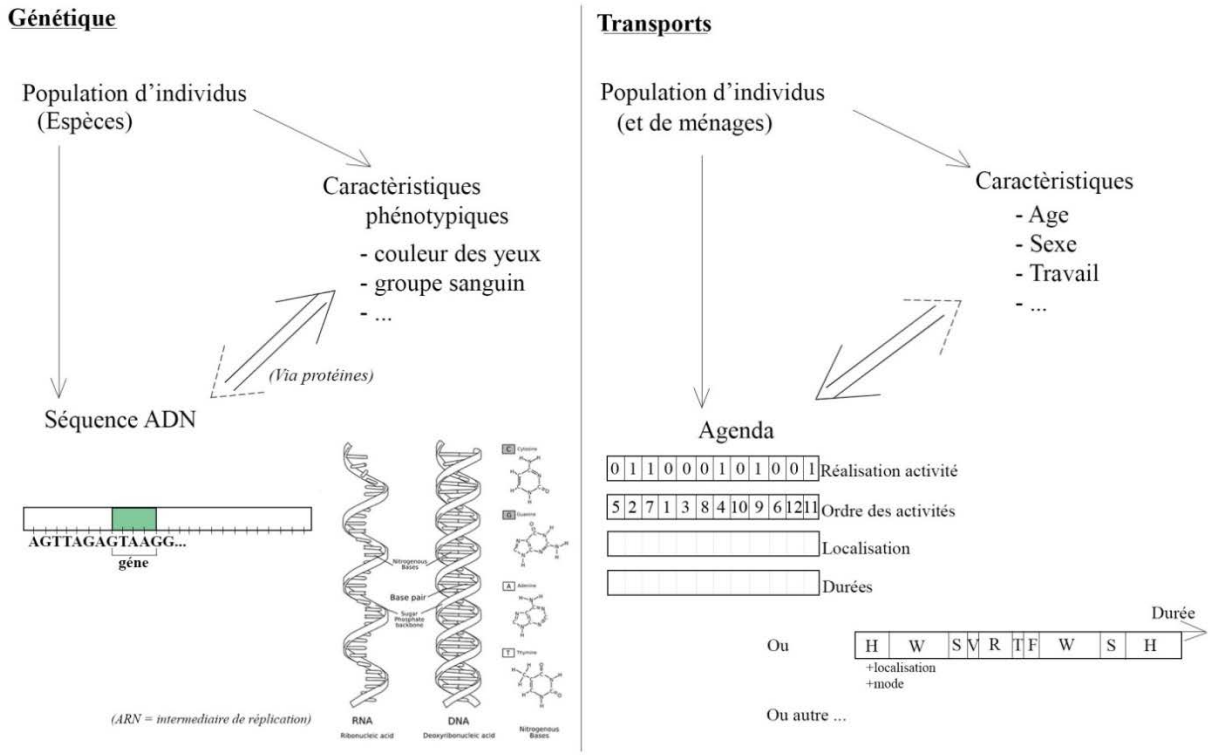


Figure 28 Schéma de réflexion sur les analogies entre génétique et transports

Ainsi, pour un problème donné, il sera possible de transposer un objet par un autre tandis que pour un autre problème voire peut-être parfois pour le même problème, une autre analogie pourra être établie. Par ailleurs, l'établissement d'agendas passe par la mise en place de processus plus ou moins complexes et il sera parfois possible de trouver un outil ou une ressource technique dans le domaine de la génétique ou de la biostatistique ou encore de la bioinformatique pour trouver une solution à un point particulier. En effet, dans la mesure où les généticiens, comme les chercheurs en transports, travaillent avec de grands ensembles de données, généralement organisées sous forme de séquences, les outils utilisés par les uns peuvent également être utiles aux autres. Nous en avons déjà montré quelques exemples dans le chapitre 2.

Mon objectif à partir de maintenant sera d'établir deux analogies afin d'appliquer les mécanismes mis à jour dans les théories de l'évolution à deux problématiques distinctes du domaine de la modélisation de la demande de transport.

Dans un premier temps, je vais proposer une démarche pour créer une population synthétique en faisant intervenir des croisements et des mutations dans une population d'enquête afin d'introduire une plus grande diversité dans la population synthétique. Puis, dans un second temps, je vais proposer une étape d'initialisation pour un algorithme génétique tel que celui présenté par Charypar et Nagel.

Dans le premier cas, la population considérée est une population humaine et nous prendrons pour leur ADN les attributs utiles dans le cadre de la modélisation de phénomènes de transports (âge, sexe, travail, possession de véhicule ...). Après avoir été répartis au sein de différentes classes, les individus issus de données d'enquêtes seront associés par couples pour créer des générations d'individus fils étendant la diversité de la population synthétique.

Dans le second cas, le but est, comme signalé précédemment d'étendre la démarche suivie dans les méthodes qui implémentent un algorithme génétique et de proposer un procédé afin de créer une population initiale, non plus aléatoire, mais construite. Dans ce cas, la population sera la population synthétique d'agents. Nous substitueront les agendas aux ADNs et nous considérerons les attributs des agents comme des traits phénotypiques fortement corrélés avec les agendas des individus.

3.2. Bases de données utilisées

La zone considérée pour la mise en application est la commune de Liège de code 62063.

3.2.1. Données de recensement

Les données par secteurs de recensement sont accessibles est disponibles en libre accès sur le site : http://census2011.fgov.be/download/statsect_fr.html. Elles sont répertoriées dans des classeurs Excel. On y retrouve entre autre les répartitions de la population par âge, sexe, types de ménages, niveaux d'étude, ...

3.2.2. Données d'enquête

Les données d'enquête proviennent d'une enquête, menée sur l'ensemble de la Belgique, qui répertorie les informations de 14582 individus. Les données renseignées concernent les individus eux-mêmes (âge, sexe, profession, type de logement...), les ménages (tailles des ménages, nombre d'enfant, revenu,...), les déplacements (fréquence des trajets selon le mode de transport, permis de conduire, ...) et enfin des données sur les séquences d'activités.

3.3. Explication de la démarche et du codage en R

Dans la suite, je vais présenter le code en R que j'ai écrit. Le but de ma démarche n'est pas de développer un modèle abouti mais démontrer la possibilité d'une mise en application des approches inspirées des mécanismes biologiques que je tâche d'échafauder.

Le langage de programmation utilisé est le langage R qui est un langage dédié aux statistiques et à la science des données. Ce langage apparu au début des années 2000, est l'un des plus utilisés par les analystes et se prête parfaitement au travail que nous voulons réaliser.

Le programme est organisé en 4 modules successifs qui permettent, à partir des données de recensement et des données d'enquête, de créer une population synthétique d'agents (modules 1, 2 et 3) puis de générer, pour un agent de la population synthétique, une population initiale d'agendas (module 4.) en vue d'appliquer un algorithme génétique.

3.3.1 Module 1. Création d'une population synthétique de ménages

Pour créer la population synthétique, nous disposons de deux sources de données : les données de recensement et les données d'enquête. Les données de recensement sont disponibles pour l'ensemble de la population mais elles sont agrégées et non pas associées à des individus ni même à des groupes d'individus. Les données d'enquêtes dont nous disposons sont des données ponctuelles mais qui fournissent des informations sur des individus membres de ménages.

La démarche la plus classique consiste à recourir à une IPF (cf. R. Beckman, 1996) afin de copier les ménages ou des groupes de ménages (ou d'individus) afin de créer une population synthétique dont les caractéristiques globales sont similaires à celles de la population réelle.

Mon idée est de considérer la population enquêtée comme une population initiale et d'appliquer des cycles de 'reproduction' afin de créer une succession de générations filles d'individus et ainsi, comme E. King le fait pour une population de drosophiles, étendre la diversité des profils d'individus servant de base à la création d'une population synthétique.

Toutefois, les individus d'enquête sont répartis en ménages, or, il n'est pas évident de créer des générations filles de ménages et cela n'aurait pas de sens de considérer les individus comme des entités 'libres'. En outre, garder la structure de ménages est intéressant dans le cadre la modélisation basée sur les activités. Ainsi, l'idée sera dans un premier temps d'isoler les ménages et de créer un 'population synthétique de ménages'. Pour cela, je propose d'utiliser un IPF. Il ne m'a pas semblé utile et ni forcément pertinent dans notre cas d'étendre la variabilité des données d'enquêtes pour les ménages.

Une fois la population synthétique de ménages créée (module 1), nous créerons les générations d'individus (module 2) puis nous sélectionnerons des individus afin de les attribuer aux ménages constituant ainsi une population synthétique d'agents.
Ainsi, le but dans le module 1 est d'extraire les données relatives aux ménages puis de récupérer les données de recensement associées afin de définir les marges pour appliquer un IPF et créer, par copie des ménages existants, la population synthétique de ménages.

En termes de vérifications, à cette étape nous calculons les écarts types entre les valeurs cibles et les valeurs correspondantes obtenues pour la population synthétique. Après modélisation, l'écart type maximal observé pour la population générée est de 0,03118968 soit 3,11 %. Cette valeur, sous le seuil des 5%, est atteinte pour l'écart type calculé pour le « nombre de ménages de 5 personnes ». Cette valeur cible est relativement faible et donc les écarts type

sont d'autant plus grands pour de faibles variations. Pour le reste de la table l'écart type est inférieur à 1,7 %. Ainsi, dans l'ensemble la procédure suivie permet de créer une population synthétique de ménages satisfaisante.

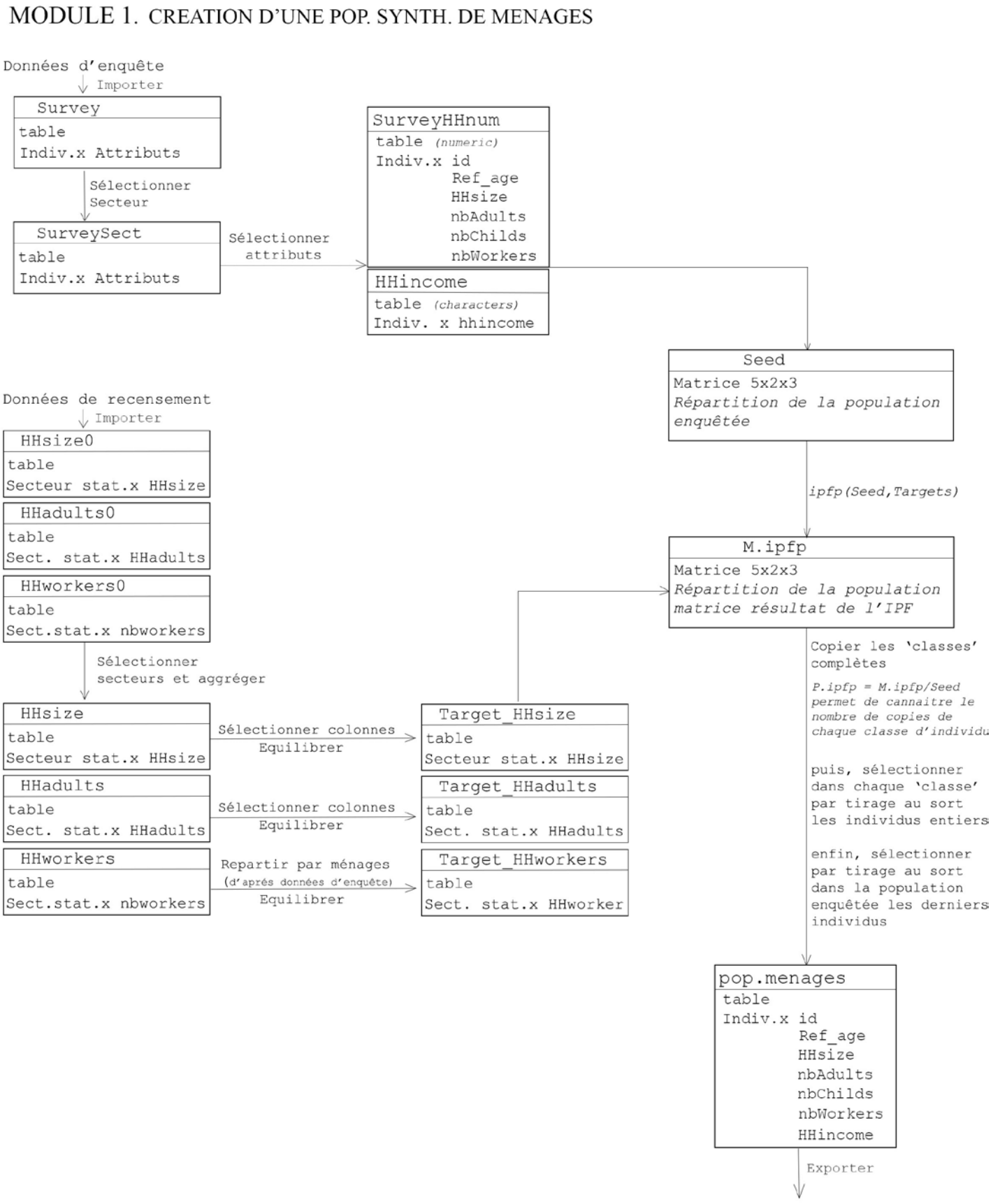


Figure 29 Module 1

On pourrait envisager une amélioration dans le processus en appliquant un algorithme pour croiser et muter les données de ménages afin d'étendre la variabilité au sein de la population de ménages. En outre, il pourrait être intéressant de créer des 'individus' pour les valeurs non représentée. Par exemple les ménages d'enquête comptaient 5 individus maximum or les données de recensement comptent l'existence de ménages de 6, 7, 8, et jusqu'à 11 individus et plus dont nous avons négligé l'influence.

3.3.2. Module 2. Etendre la variabilité des individus d'une population synthétique

Le module 2 prends en données d'entrée les données d'enquête et crée par croisements et mutations de nouveaux individus qui serviront à créer la population synthétique d'agents.

La première étape de l'algorithme est la sélection des individus selon le secteur d'étude pour lequel on souhaite créer une population synthétique. Dans mon cas le secteur d'étude est la commune de Liège (identifiée par le code 62063). Ensuite, nous sélectionnons les données utiles dans le cadre de la modélisation que l'on souhaite réaliser et nous classifions les individus.

Le processus de croisement va permettre de recombiner les valeurs d'attributs des individus d'enquête pour en créer de nouveaux. Or, parmi les attributs certains sont corrélés et ces corrélations ne peuvent pas être ignorées si l'on ne veut pas voir apparaître des profils improbables comme par exemple des retraités de 20 ans. La création de classes est une première façon de gérer le problème de corrélations.

Ainsi, je recours à deux critères afin de créer 3x3 classes d'individus. Le premier critère est la position au sein du foyer : "Reference Person HH", "Partner" ou "Child". Ce critère sera prépondérant pour la répartition des individus dans les ménages. Le second critère est le "statut" : "Scolaire/étudiant", "actif occupant un emploi", "Retraité, chômeur, personne au foyer, incapacité, ...".

Les caractéristiques restantes parmi les attributs que j'ai retenus pour mon modèle sont : l'âge, le sexe, le niveau d'étude (diplômes), le type d'emploi et la mention de compagnon et enfants. Parmi ces critères il reste une corrélation entre le niveau d'étude et le type d'emploi. De ce fait dans notre modèles ces deux caractéristiques seront liées et ne seront pas croisées entre elles ; c'est la deuxième façon de régler le problème de corrélations.

On pourra remarquer que dans mon cas, j'ai peu d'attributs ainsi s'il est laborieux de définir les corrélations (ce que je fais par ailleurs ici de façon arbitraire en recourant à l'observation et au bon sens) et de définir un moyen de résoudre le problème, cela reste facilement réalisable dès lors que l'on dispose de peu d'attributs. Si l'on ajoute des attributs décorrélés cela ne pose pas de problèmes, mais si l'on ajoute plus d'attributs corrélés, il faudra peut-être aussi envisager de créer plus de classes, le risque est alors d'avoir un nombre réduit d'individus fortement semblables dans chaque classes et de limité l'efficacité et l'intérêt de la méthode proposée.

Une autre façon de procéder pourrait alors être de travailler avec des probabilités traduisant les corrélations. Une première idée pourrait être d'utiliser ces probabilités lors des étapes de croisement et de mutation toutefois cela risque d'une part de complexifier la procédure et d'autre part cela pourrait contraindre la formation d'une réelle diversité. Ainsi, l'idée serait plutôt de déterminer une probabilité a posteriori pour chacun des individus et d'éliminer les individus improbables. Par ailleurs, ces scores de probabilité pourraient servir lors de la sélection des individus pour le remplissage des foyers à l'étape suivant. La difficulté est alors de déterminer les probabilités associées aux corrélations entre attributs.

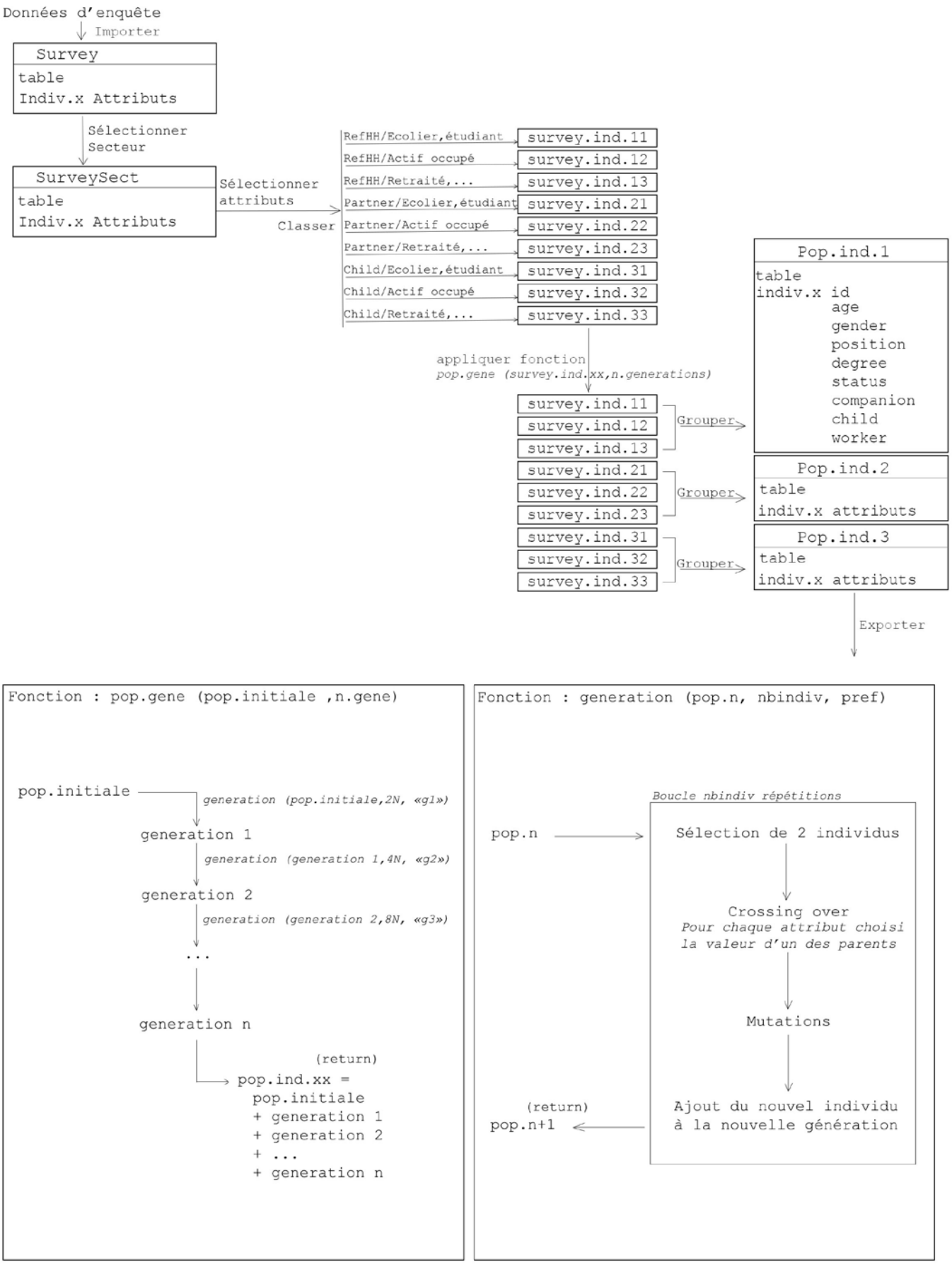


Figure 30 Module 2

Si l'on revient à notre modèle, après avoir déterminé des classes et les attributs corrélés, pour chaque classe une population de descendants est créée en mimant le protocole de création d'une population synthétique de drosophile mis en place par E. King.

Les individus d'enquête d'une classe sont sélectionnés deux à deux pour créer par croisement puis mutation un nouvel individu de la génération F1 jusqu'à ce que le nombre d'individus visé soit atteint. Dans mon modèle, le nombre d'individus de la génération fille est un paramètre à déterminer, et je propose de créer des générations filles comptant le double d'individus par rapport à la population précédente. Par ailleurs deux parents sont sélectionnés pour créer un nouvel individu et pour chaque nouvel individu, les parents sont tirés au sort parmi l'ensemble de la population de la génération précédente. On pourrait imaginer d'autres façons de procéder, par exemple, une possibilité est de former des couples de parents 'sans remise' pour former systématiquement deux nouveaux individus l'un héritant des caractères dont l'autre n'hérite pas et créer ainsi des générations filles comptant autant d'individus que la génération précédante.
Comme cela est fait pour les drosophiles, la génération F2 est engendrée par les individus de la génération F1 exclusivement et ainsi de suite.
A la fin les individus de toutes les générations sont rassemblés pour former une population d'individus candidats pour l'affectation à un ménage et donc candidat pour intégrer la population synthétique d'agents.

3.3.3. Module 3. Création d'une population synthétique d'individus

Le module 3 se compose de deux parties. Il s'agit dans un premier temps d'assigner à chacun des ménages de la population de ménages (créée dans le module 1) des individus sélectionnés dans la population étendue créée le module 2. Une première sélection des individus est réalisée afin de sélectionner les individus ayant un profil qui correspond aux attributs du ménages ; les critères de sélection sont la 'position' (référent, partenaire ou enfant), la classe d'âge, l'emploi, le critère « en couple » (o/n), « avec enfants » (o/n). Une fois cette première sélection effectuée, l'individu retenu est sélectionné au hasard parmi les individus restants.

La seconde étape vise à ce que la population synthétique ait des caractéristiques qui correspondent à celles de la population réelle en termes de répartition par âges (classes d'âge de 5ans), par sexe, selon le type de diplôme, le type d'emploi.... Dans la mise en application de mon modèle je n'ai retenu que deux paramètres : la répartition des âges et la répartition selon le sexe, néanmoins, la méthodologie appliquée permet d'intégrer autant de caractéristiques de la population que nécessaire.

Le principe pour cette deuxième étape est d'appliquer une méthode de Hill Climbing. Pour chaque agent de la population synthétique un score est calculé afin de déterminer la contribution à l'éloignement par rapport aux valeurs cibles de chaque critère. En fait, on fait intervenir deux scores :

$$S_1 = \sum_i si(\, nbindividus.critere_i - cible.critere_i > 0\ ;\ 0\ ;\ 1)$$

Le premier score détermine pour combien de critères l'individu contribue à faire en sorte de la population synthétique soit en excès d'effectifs par rapport à la population réelle. Plus le score est élevé plus l'individu devra être remplacé en priorité.

$$S_2 = \sum_i (nbindividus.critere_i - cible.critere_i)$$

Le second score est une valeur qui peut soit être positive, dans ce cas l'individu contribue à faire en sorte que les classes soient en excès d'effectifs par rapport à la population réelle, si cette valeur est négative cela traduit le fait que l'individu appartient plutôt à des classes en déficit d'individus.

Les individus à remplacer sont déterminés à l'aide de ces deux scores en commençant par les individus de pire score. Lorsqu'un individu est identifié pour être remplacé, une sélection de candidats pour le remplacer est effectuée parmi les individus provenant du module 2 (pop d'enquête étendue). Après une sélection des candidats éligibles sur les critères d'âge, de compagnon, d'enfants, d'emploi, etc., le remplaçant est tiré au sort parmi les meilleurs retenus (selon les scores). La population synthétique est alors mise à jour, les scores de la population sont recalculés et un nouvel individu à remplacer est identifié. Le processus est ainsi répété jusqu'à ce que les caractéristiques de la population synthétique correspondent à celles de la population réelle.

On remarque qu'à mesure que le processus converge si l'on ne sélectionne que des individus de classes excédentaires, leurs profils ne permettent pas systématiquement de les remplacer par des candidats représentant les classes les plus déficitaires. Pour illustrer notre propos imaginons qu'une classe d'âge (30-35 ans par exemple) est en excès d'effectifs, un individu de cette classe devra alors être sélectionné afin d'être remplacé par un individu appartenant de préférence à une classe d'âge en déficit d'effectifs. Or, il est possible de sélectionner des individus à remplacer ayant des profils (sexe, mariage, enfants,...) qui ne permettent pas de trouver un "bon" remplaçant. Afin d'améliorer le processus, on ne sélectionne pas seulement de "mauvais" individus afin de les remplacer mais aussi de "bons" individus. La démarche est alors la suivante. Lorsqu'un bon individu est sélectionné cela signifie qu'il appartient à des classes à compléter. On recherche alors les candidats que l'individu sélectionné pourrait éventuellement remplacer et on sélectionne parmi ces individus l'un de ceux qui ont les plus mauvais scores. Cet individu devient alors l'individu à remplacer et on applique le processus décrit précédemment sachant qu'il existe de bons remplaçants pour cet individu.

Dans notre cas, avec deux critères, on observe une convergence relativement rapide. En fait, la répartition selon le sexe est proche des valeurs cibles et la convergence pour ce critère est observée après quelques itérations à peine. Il ne reste alors plus que le critère âge à vérifier, chaque remplacement peut donc réduire les écarts entre valeurs observées dans la population synthétique et valeurs cibles de 2 (1 pour la classe en excès qui est déchargée d'un individu et 1 pour la classe en déficit qui voit son effectif augmenté d'un nouvel individu).

Lors de la mise en application, la somme des écarts en valeur absolue entre les effectifs de chaque classe d'âge de la population synthétique initiale et les valeurs cibles d'effectifs représente une valeur 2044. D'après l'observation faite précédemment avant même de commencer un minimum de 1022 remplacements devront être effectués. En pratique, on a observé la convergence après 1863 itérations.

La meilleure façon d'améliorer le système est donc de travailler sur l'efficacité de la première étape afin de produire une population synthétique plus proche de la population réelle dès le départ. Pour ce faire, une possibilité est d'ajouter des probabilités lors de la sélection des individus après que la sélection d'après critères a été effectuée.

Une fois la population synthétique formée, il faudrait mettre en place et appliquer une procédure de vérification du modèle. Je ne vais pas effectuer de vérification, néanmoins, je vais proposer une démarche de vérification.
Une façon de procéder serait de prendre 30 à 50 % des individus de la population d'enquête afin de générer via le programme une population synthétique qui vise à reproduire toute la

population d'enquête. Cela permettrait de vérifier les performances du modèle et de déterminer si, à partir d'un échantillon d'individus, il est capable de produire une population synthétique qui reflète l'ensemble de la population.

Notons que l'échantillon doit être un minimum représentatif, ce qui est une condition sine qua non lorsqu'on veut produire une population synthétique à partir d'un panel d'individus.

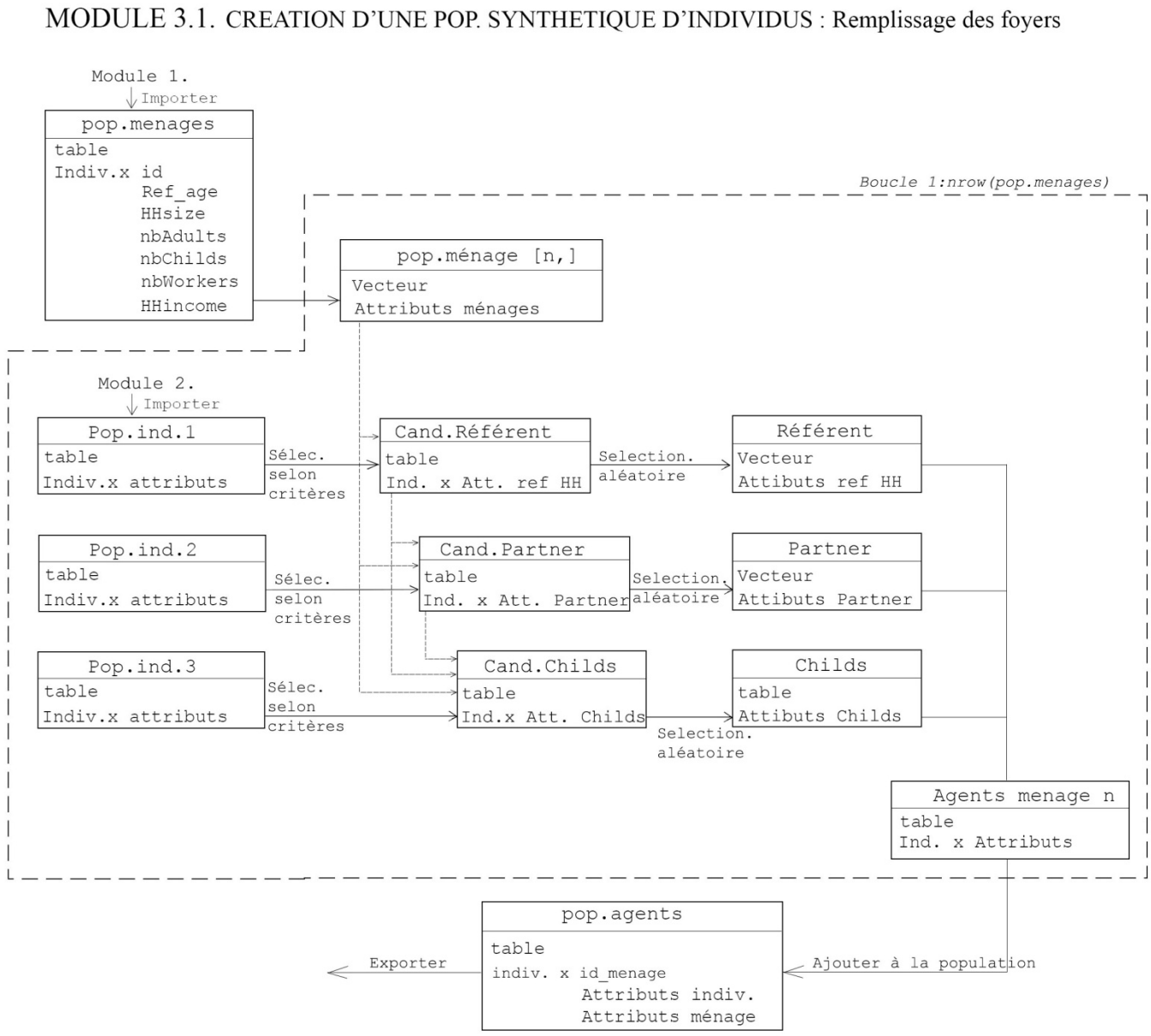


Figure 31 Module 3.1.

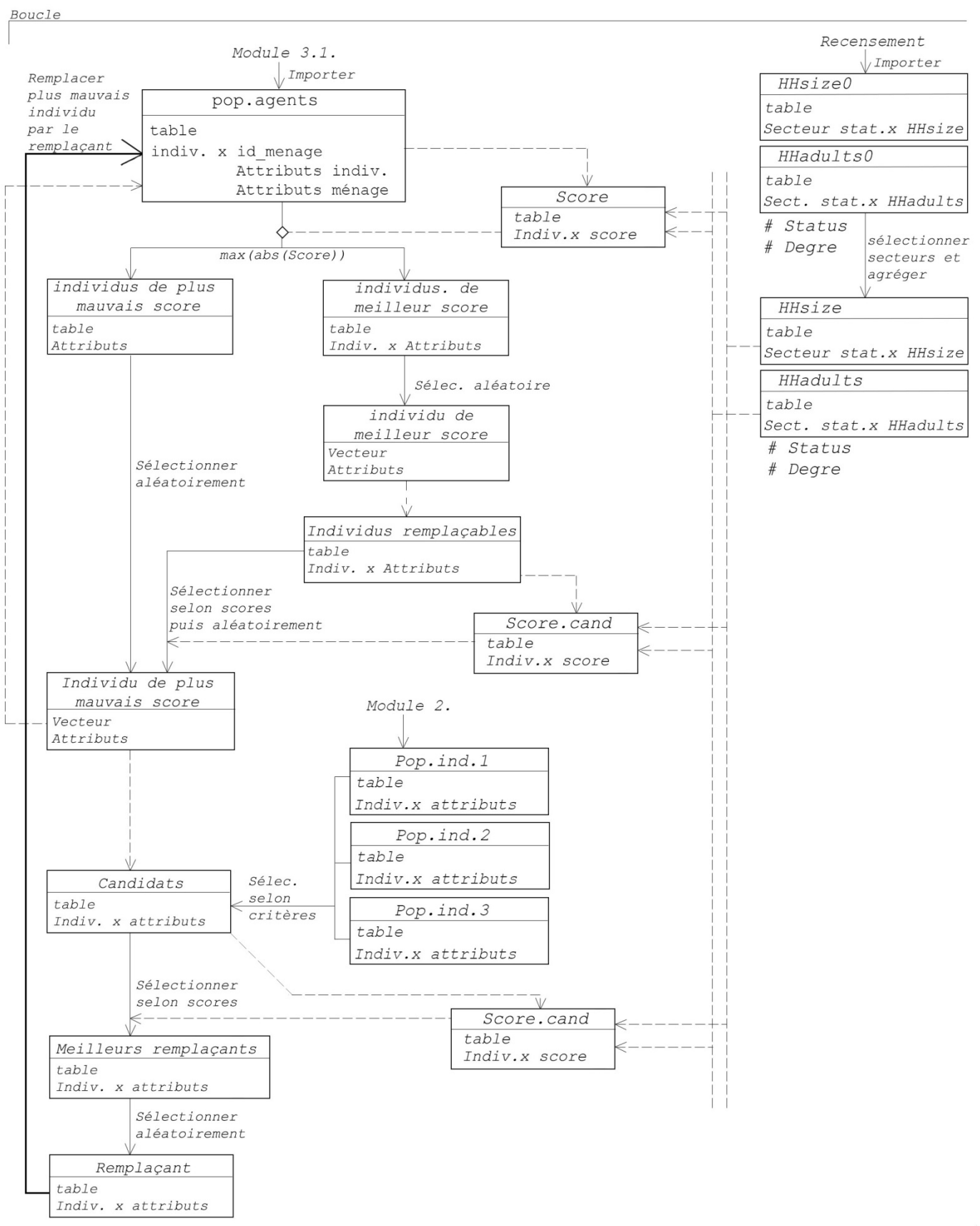


Figure 32 Module 3.2.

3.3.4. Module 4. Création d'agenda : création d'une population initiale pour un algorithme génétique

Une fois la population synthétique créée, l'étape de création des agendas peut commencer. Dans ce module le but est de proposer une méthodologie pour générer des populations de solutions construites et non plus aléatoires comme populations initiales d'un algorithme génétique tel que celui présenté par Charypar et Nagel (2003).

L'idée de départ est d'établir un parallèle entre le couple ADN/phénotype et le couple Agenda/caractéristiques d'un individu. En génétique l'ADN d'une personne détermine grandement ses caractères phénotypiques. Par exemple, la couleur des yeux d'un individu dépends de ses gènes ainsi connaître les allèles des gènes qui déterminent la couleur des yeux permet de savoir quelle est la couleur des yeux d'un individu sans voir cet individu, et à l'inverse un individu avec les bleus aura forcément des gènes qui expriment ce caractère. Il y a ainsi une sorte d'équivalence entre phénotype et génotype.

Dans un modèle de transport, si l'on voit l'agenda à la place de l'ADN et les attributs liés aux comportements de transport comme le phénotype de l'individu, on peut affirmer dans un premier temps que l'agenda d'une personne est en grande partie déterminé par son profil, c'est à dire ses caractéristiques. Il existe donc des corrélations entre les caractéristiques d'un individu et son agenda. Mais pourrait-on parler d'une "équivalence" entre agenda et attributs d'un individu ? Si cela est le cas, on peut établir un parallèle entre génétique et transports, et en particulier affirmer l'analogie entre ADN et agenda. Dans ce cas, un enfant qui hérite à la fois des gènes de son père et de sa mère aura des traits de caractère physique semblables à ceux de ses parents, dans notre modèle de transport on procédera alors en sens retour et nous prétendrons qu'un individu qui a des caractéristiques proches de celles de deux autres individus que nous qualifierons de "parents" aura un agenda qui rappelle également celui de ses parents. Ainsi, la démarche adoptée pour créer l'agenda consistera d'abord à trouver, dans la population d'enquête, deux individus proches de l'individu pour lequel nous voulons créer un agenda. Pour ces deux individus « parents », nous disposons de données sur leurs séquences d'activités. Ces parents permettront alors de générer un nouvel agenda pour l'individu 'fils'.
Finalement cela revient à considérer implicitement les corrélations entre les caractéristiques d'un individu et son agenda : deux individus présentant des profils semblables ont de fortes de chances d'avoir également des plannings d'activités (puis plus largement des agendas) semblables.

La question à laquelle nous devons maintenant répondre est alors, y a-t-il équivalence entre l'agenda et les caractéristiques d'un individu ? Nous avons déjà établi le sens aller : un agenda est déterminé par les attributs propres à un individu. En fait, tout comme pour le couple phénotype/génotype l'équivalence n'est pas exacte, certains facteurs environnement ont une influence, à commencer par les influences des autres membres d'un même ménage qui ont des caractéristiques propres et donc des agendas propres. Parmi les facteurs environnementaux, il est également possible de citer les horaires d'ouvertures ou encore l'accès à certains types d'activités. Néanmoins, si les caractéristiques renseignées pour chacun des individus sont suffisamment complètes et pertinentes, elles pourront permettre de définir un agenda avec une bonne précision

Maintenant, pour ce qui du sens retour de l'équivalence, cela correspond à se demander si à partir d'un agenda, il serait possible de déterminer les caractéristiques d'un individu. Exprimer comme cela, il semble qu'en effet dans la une certaine mesure, selon le degré de précision des agendas et leur "forme" il sera possible de retrouver certaines (pas forcément toutes) des caractéristiques d'un individu. Ainsi, il est possible d'établir que oui, dans une certaine

mesure, il existe une équivalence entre agenda et caractéristiques d'un individu. Ajoutons de plus, que le sens retour nous intéresse pour établir le modèle et appuyer l'analogie, mais que ce n'est que le sens aller, plus précis, qui sera appliqué.

Finalement, l'analogie pourra être validée si tant est que les caractéristiques renseignées pour les agents du modèle de transport sont suffisamment exhaustives et pertinentes. En fait dans une certaine mesure, plus on disposera d'informations sur les individus plus la méthode sera pertinente, l'enjeu étant ensuite de définir la notion de degré de ressemblance entre individus.

Que signifie que deux individus se ressemblent dans le cas qui nous occupe ?
Il s'agira de mettre en évidence des attributs qui ont plus de poids que d'autres dans les comportements, par exemple le fait de travailler ou non est une information majeure, le type de travail aura une importance d'ordre secondaire, la distance au lieu de travail est un autre exemple de donnée tout à fait pertinente. Par ailleurs pour certains critères, des valeurs non identiques peuvent parfois conduire à des comportements similaires (par exemple on peut l'envisager pour deux personnes qui ont des emplois différents mais qui ont au final des conditions de travail similaires).

Pour ma part, j'ai défini une fonction relativement simple qui attribue un score de ressemblance entre deux individus selon l'identité entre certaines caractéristiques (avec pondération, cf. le code en annexe). Cela a été fait en recourant à l'observation et au bon sens. En toute rigueur la fonction devra être établie et calibrée à partir de données statistiques chiffrées.

Par la suite, en accord avec ce que nous venons de dire, le principe serait donc de sélectionner deux parents dans la population d'enquête servant à générer par croisement et mutation un nouvel agenda pour un individu fils. Les deux parents sont des individus qui présentent des scores de ressemblance élevé avec l'individu 'fils'.

Une fois l'agenda créé il faut déterminer sa vraisemblance : est-il réalisable ? Et est-il satisfaisant ? Et s'il ne l'est pas il faut alors réitérer l'opération jusqu'à trouver un 'bon' agenda. A ce stade deux problèmes apparaissent. Premièrement, comment déterminer si un agenda est acceptable ? Deuxièmement, étant donnée la méthode de travail employée, il semble nécessaire d'ajouter quelques conditions lors de la création des agendas pour obtenir un candidat acceptable.

Dès lors, si au lieu de générer un seul agenda, nous générions une population d'agenda servant de point de départ à un algorithme génétique, l'algorithme génétique permettrait de résoudre l'ensemble des problèmes qui apparaissent et d'améliorer le résultat. En effet, c'est la fonction d'adaptabilité qui permet dans ce cas de figure de déterminer la "performance" des agendas. En ce qui concerne la condition d'arrêt, elle équivaut désormais à la convergence de l’algorithme. Enfin, il n'y a plus nécessité d'ajouter de conditions pour la création d'agenda, c'est la fonction d'adaptabilité qui permet de sélectionner les meilleurs agendas et d'écarter les cas absurdes. En outre, cela permet, non plus de se contenter d'un agenda dont on juge arbitrairement s'il est suffisamment acceptable, mais de trouver un agenda optimal.

Les agendas ne sont alors plus générés à partir de deux parents seulement mais à partir d'un groupe d'individus "parents". L'initialisation de l'algorithme avec une population construite de la sorte à deux conséquences envisageables. La première est une convergence plus rapide, la seconde est la possibilité plus élevée d'atteindre un optimum local et non pas global. Toutefois, si cette seconde conséquence semble problématique, elle ne l’est pas forcément.

MODULE 4. CREATION D'AGENDA : création d'une population initiale pour un algorythme génétique

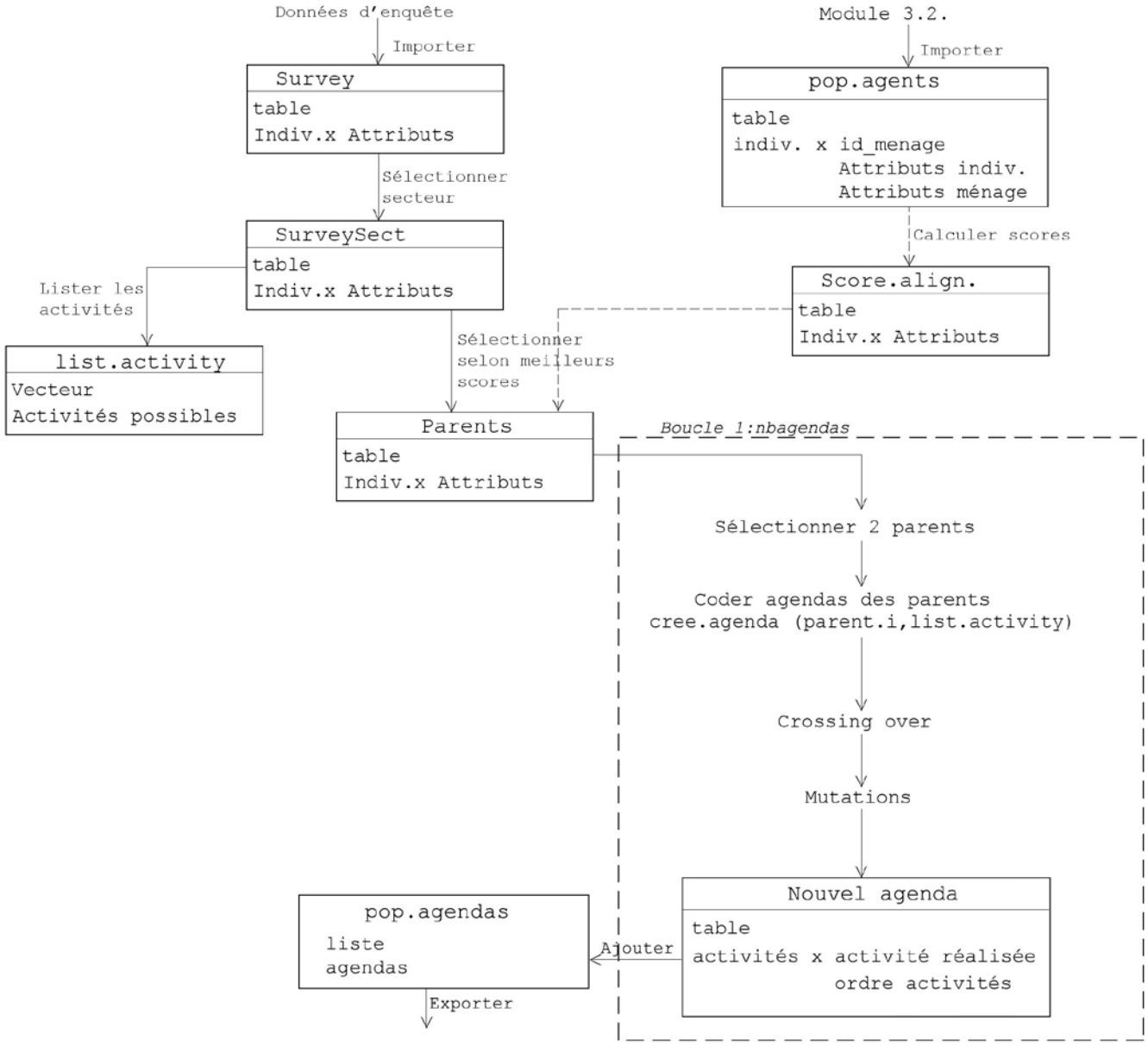


Figure 33 Module 4

En effet, Charypar et Nagel (2003) l'affirment eux-mêmes lorsqu'ils proposent de recourir à des algorithmes génétiques : les agendas réels sont loin d'être optimaux et l'objectif n'est pas forcément de trouver la solution optimale au problème mais de trouver une solution satisfaisante.

J'irai même plus loin en affirmant que si un optimum local est atteint c'est qu'il décrit très probablement la réalité des comportements humains dans la mesure où il est trouvé dans l'espace de solutions généré à partir de la population initiale d'agendas qui se base sur les séquences d'activités réelles observées lors des enquêtes.

Tous les points essentiels de ce module ont maintenant été discutés à l'exception de la façon de représenter les agendas. La méthode de codage mise en place par Charypar et Nagel et également reprise par K. Meister est une représentation des agendas adaptée à la mise en œuvre de phénomènes de croisement et de mutation. Ainsi, je reprends à mon tour la même

structure pour représenter les agendas. Néanmoins, ne disposant pas d'informations sur les durées des activités ni sur les localisations, dans mon cas les agendas se composent deux vecteurs uniquement. Un vecteur binaire qui détermine si une activité est programmée ou non et un vecteur d'entiers qui détermine l'ordre des activités. Dans mon modèle, seules les activités programmées sont ordonnées les autres ont une valeur nulle, puis à la fin du module 4 une permutation globale est établie afin de se ramener à la forme envisagée par Charypar et Nagel tout en conservant l'ordre effectif des activités réalisées. Par ailleurs, l'absence des notions de durée et de localisation dans les agendas n'est pas réellement un problème dans la mesure où ce sont des aspects secondaires au vu de la démarche proposée. Les localisations, de même que les durées sont à définir selon l'individu auxquels elles se rapportent et selon les séquences d'activités, donc en aval de création des séquences d'activités.

Pour ce qui est des localisations, il est indispensable de distinguer des localisations "fixes" qui ne dépendent pas du contexte de réalisation des activités (ce sont les localisations du lieu de résidence ou du lieu de travail par exemple) et les localisations "flexibles" qui peuvent être définies a posteriori une fois que la séquence d'activités est définie (comme les lieux de repas, de shopping, de loisir, de promenade, ...). Le premier type de localisations dépend de l'individu et peut être défini et pris en considération pour déterminer les séquences d'activités (exemple : distance Work-Home), le second type de localisations peut permettre d'invalider la faisabilité ou de réduire l'utilité d'un agenda mais doit être déterminé après que la séquence d'activités n'ait elle-même été déterminée.

La notion de durée des activités est un peu plus complexe à intégrer au modèle. Elle peut être renseignée par enquête ou être associée aux activités elles-mêmes. La temporalité de chaque activité peut être définie postérieurement à l'établissement des séquences d'activités. Il s'agit alors de déterminer, activité par activité, les temps de trajets et les temps de réalisation des activités quitte à effectuer une sélection parmi certaines activités non obligatoires qui ne peuvent pas être réalisée par manque de temps.

Enfin, l'ajout des modes de transport ne semble pas poser problème outre mesure.

3.3.5. Et ensuite ?

Une fois la population initiale d'agendas produite, il s'agit d'appliquer un algorithme génétique tel que celui décrit par Charypar et Nagel puis de vérifier les résultats et la performance du processus. Cela donnerait lieu à un module 5. que je ne code pas dans le cadre de ce travail car ce n'est pas d'un intérêt majeur dans le cadre du sujet qui m'occupe dans la mesure où il n'y a spécifiquement de mise en application de techniques ou de mécanismes génétiques ou biostatistiques. Une fois que la population synthétique est établie et que chacun des agents est doté d'un agenda, un modèle de micro simulation peut être utilisé pour déterminer les différents flux de voyageurs.

3.4. Résultats et conclusions

A travers ce travail de modélisation, ont été présentées deux méthodologies inspirées des travaux menés par E. King sur la création de population de drosophiles et plus généralement sur les mécanismes biologiques mis en œuvres dans le processus de sélection naturelle.
La première méthodologie proposée permet de créer une population synthétique d'individus variés à partir de profils d'individus issus d'enquêtes.

La seconde méthodologie s'attaque à la problématique de la création d'agendas en proposant une méthode d'initialisation pour des algorithmes génétiques tels que celui présenté par Charypar et Nagel (2003).

Pour chacun des cas, un modèle épuré relativement simple a été codé en langage R et permet d'obtenir des résultats a priori cohérents, démontrant qu'il est possible de trouver une ressource exploitable pour développer de nouveaux modèles et/ou améliorer des modèles existants pour l'analyse et la gestion de la demande de transport.

On peut reprocher à ce travail un manque de vérifications, de calculs de performances et de comparaisons à d'autres modèles existants. Malgré tout, soulignons que l'enjeu premier est d'établir et de mettre en œuvre une démarche qui se fonde sur un raisonnement par analogie utilisant comme ressource à la l'innovation, les travaux et les observations menés dans le domaine de la biologie dont les objets ont des structures et des interactions qui rappellent celles mises en œuvre en transport dans les modèles basés sur l'activité. Dans cette optique, il me semble que l'exemple développé permet de faire ressortir cette proximité entre les domaines.

Par la suite d'autre part, il serait intéressant de s'intéresser à l'utilisation d'outils, de méthodes et de techniques statistiques pouvant être appliquées afin de traiter un problème ou un autre dans le domaine des transports. Quelques pistes de recherches ont déjà été mises à jour dans le chapitre 2.

Par ailleurs, notons que la ressource DSPR n'a pas été créée pour elle-même, mais afin de servir de base de travail pour les généticiens qui étudient des populations de drosophiles. Les divers travaux qui utilisent la ressource DSPR afin de décrire les mécanismes et les interactions entre des différentes unités du « vivant » constituent une ressource qui ne manque pas d'intérêt. Parmi ces travaux on peut notamment évoquer les travaux sur la dissection de traits complexes, ou la recherche de loci de gènes responsables de caractères particuliers (résistance à la nicotine, vulnérabilité à la chimiothérapie, ...).

Conclusion

La modélisation des phénomènes de transport par des modèles basés sur l'activité est un problème complexe qui fait intervenir des connaissances dans de multiples domaines : la planification des infrastructures de transport, la planification urbaine, l'informatique, la politique, l'écologie, etc. Mais toute science se nourrit d'autres sciences, ainsi, d'autres domaines d'études a priori sans rapport direct avec la demande de transport peuvent servir à développer de nouveaux modèles ou à améliorer les modèles existants. En particulier, des travaux issus des domaines de la biologie, de la biostatistique et de la bioinformatique peuvent inspirer les chercheurs dans la création de modèles de prévision de la demande de transport, non pas de façon directe, mais via des approches par analogie et la recherche de similitudes.

Si le raisonnement par comparaison et analogie est un procédé relativement naturel pour l'Homme, il n'est pas toujours évident à mettre en place. En effet, raisonner par analogie revient à examiner les points communs, les différences et les relations entre plusieurs concepts. En outre, Sternberg identifie quatre étapes au raisonnement par analogie : l'encodage (identifier et représenter les entités en jeu), l'inférence (identifier les relations entre les entités), la mise en correspondance (identifier les relations de second ordre) et l'application (exploitation, résultats, ...).

Cela implique donc, dans le cas qui nous occupe, de se doter d’une certaine connaissance du domaine de la biologie, d'en comprendre les enjeux et d'accepter de se confronter aux problématiques des biologistes avant de pouvoir comprendre les solutions apportées par ces derniers. Il faut ainsi être capable de se les approprier dans un domaine tiers. Cela demande également une certaine ouverture d'esprit, une capacité de prendre de la distance et parfois même de remettre en cause les modèles existants afin de trouver de nouvelles perspectives et des alternatives aux modèles et sous-modèles existants.

Enfin, cela demande de faire preuve de souplesse. D'une part de souplesse d'esprit afin d'aborder différents problèmes sous différents points de vue et d'autre part, de faire preuve de souplesse dans l'évaluation du raisonnement par analogie. Le raisonnement par analogie comprend une part de subjectivité qui oriente la description et la méthode de résolution du problème posé. Il doit servir de base de travail afin d'élaborer de nouvelles approches, de nouveaux outils, ou de nouvelles méthodologies, mais il conviendra de s'en détacher pour étudier les processus plus élémentaires et refonder les modèles dans notre domaine d’étude.

Dans le second chapitre, une variété de travaux menés dans le cadre de la recherche en biologie ont été présentés ainsi que quelques pistes de réflexion quant à la manière de les exploiter pour les appliquer à des travaux dans le domaine des transports. Parmi les travaux menés en biologie tous n'aboutissent pas sur des résultats ou des techniques utilisables. D'autre part, parmi les travaux exploitables, force est de constater que tous ne sont pas aussi facilement abordables et tous ne sont pas aussi porteurs. Néanmoins, l'exemple développé dans le troisième chapitre démontre la pertinence de la démarche qui permet d'établir des méthodologies à la fois innovantes et performantes. Type particulier de raisonnement inductif, le raisonnement par analogie représente une technique de créativité efficace qui permet de donner un souffle nouveau à la recherche de la faire progresser.

Nous avons donc investigué ici la possibilité d'utiliser des techniques et des outils de biostatistique et de bioinformatique pour la modélisation de phénomènes de transport. Il reste tout aussi envisageable d'explorer d'autres domaines d'étude afin d’y trouver d'avantages de types d'outils et de techniques transposables pour la création des modèles de la demande de transport.

Bibliographie

- *Chapitre 1*

ANTONI, J.P., AUPET J.B. et VUIDEL, G. (2011) La population synthétique localisée "Un pré-requis pour la modélisation" (support de cours), Laboratoire ThéMA, Université de Franche-Comté, Besançon.
www.mobisim.org

ARENTZE, T., TIMMERMANS, H.J.P. et HOFMAN, F. (2007) Creating Synthetic Household Populations: Problems and Approach. Transportation Research Record: Journal of the Transportation Research Board 2014: 85–91.

BECKMAN, R.J., BAGGERLY, K.A. et Mc KAY, M.D. (1996): Creating Synthetic Baseline Populations. Transportation Research, Vol. 30, No. 6, 415-429.
DOI:10.1016/0965-8564(96)00004-3

BECKMAN, R., et al. (2015) Generating a synthetic population of the United States, Virginia Tech, USA.
http://staff.vbi.vt.edu/swarup/papers/US-pop-generation.pdf

BHAT, C. R. et KOPPELMAN, F. S. (1999) Activity-Based Modeling of Travel Demand, Handbook of Transportation Science, chapter 3, Norwell, Massachusetts (USA) : Kluwer Academic Publishers.

CASTIGLIONE, J., BRADLEY, M. et GLIEBE, J. (2015) Activity-Based Travel Demand Models: A Primer, SHRP 2 Report S2-C46-RR-1, Washington, D.C. (USA) : Library of Congress. ISBN: 978-0-309-27399-2

CHAPIN, F.S. Jr. (1974) Human Activity Patterns in the City: Things People Do in Time and Space, John Wiley and Sons, London.

CIRILLO, C. et al. (2004) Les enquêtes sur les comportements de mobilité, et après ? , Reflets et perspectives de la vie économique (Tome XLIII), p. 111-121.
DOI 10.3917/rpve.434.0111

FEIL, M.(2010). Choosing the daily schedule : expanding Activity-Based travel demand modelling (Master of Science thesis), ETH Zurich, p. 1-31.

GUO, J.Y. et BHAT, C.R. (2007) Population Synthesis for the Microsimulating Travel Behavior. Transportation Research Record: Journal of the Transportation Research Board 2014: 92–101.

HÄGERSTRAND, T. (1970). What about People in Regional Science?, Paper for the Ninth European Congress of the Regional Science Association.

HÄGERSTRAND, T. (1982) Diorama, path and project, TESG, 73.

JEONG, B., LEE, W., KIM, D.-S. et SHIN, H. (2016) Copula-Based Approach to Synthetic Population Generation, PLoS ONE 11(8): e0159496.
doi:10.1371/journal.pone.0159496

MOECKEL, R., SPIEKERMANN, K. et WEGENER, M. (2003) Creating a Synthetic Population, Paper presented at the 8th International Conference on Computers in Urban Planning and Urban Management (CUPUM), Sendai, Japan.

Mc NALLY, M. G. et RINDT, C. R. (2007). The Activity-Based Approach, UCI-ITS-WP-07-1, Institute of Transportation Studies, University of California.

NAMAZI-RAD, M.-R., MOKHTARIAN, P. et PEREZ, P. (2014) Generating a Dynamic Synthetic Population – Using an Age-Structured Two-Sex Model for Household Dynamics, PLoS ONE 9(4): e94761.
doi:10.1371/journal.pone.0094761

RASOULI, S. et TIMMERMANS, H.J.P. (2014). Activity-based models of travel demand : promises, progress and prospects, The International Journal of Urban Sciences, 18(1), 31-60. DOI: 10.1080/12265934.2013.835118

ŠVEDA, M., et MADAJOVA, M. (2012) Changing concepts of time geography in the area of information and communication technologies, *Acta Universitatis Palackianae Olomucensis – Geographica*, Vol. 43, No. 1, 2012, pp. 15-30

YE, X., KONDURI, K., PENDYALA, R.M., SANA, B., et WADDEL, P. (2009) A Methodology to Match Distributions of Both Household and Person Attributes in the Generation of Synthetic Populations. Washington, U.S.A., 2009. Transportation Research Board - 88th Annual Meeting.

Activity-Based Travel Demand Models (support de cours), Ira A. Fulton School of Engineering, Arizona. fulton.asu.edu

Transportation forecasting. (s. d.). Dans Wikipédia, l'encyclopédie libre. Repéré le 30 juin 2017 à https://en.wikipedia.org/wiki/Transportation_forecasting

Bibliographie pour les modèles et auteurs de référence cités dans le premier chapitre :

Arentze, T. A., & Timmermans, H. J. P. (2000).Albatross, a learning-based transportation oriented simulation system. Eindhoven: EIRASS, Eindhoven University of Technology.

Arentze, T. A., & Timmermans, H. J. P. (2004). A learning-based transportation oriented simulation system. Transportation Research B,38, 613–633.

Arentze, T.A., & Timmermans, H. J. P. (2005).Albatross V2, A learning-based transportation oriented simulation system. Eindhoven: EIRASS, Eindhoven University of Technology.

Arentze, T. A., & Timmermans, H. J. P. (2011b). Towards a multi-agent model of activity-travel dynamics in complex urban environments. In W. Y. Szeto, S. C. Wong, & N. N. Sze (Eds.), Transportdynamics(pp. 767–774). Hong Kong: HKSTS

Ben-Akiva, M. E., & Bowman, J. L. (1998). Activity based travel demand model systems. In P. Marcotte & S. Nguyen (Eds.),Equilibrium and advanced transportation modeling(pp. 27–46). Montreal: Kluwer.

Ben-Akiva, M. E., Bowman, J. L., & Gopinath, D. (1996). Travel demand model system for the information area.Transportation,25, 241–266.

Bhat, C.R. (1997a) Recent methodological advances relevant to activity and travel behavior analysis, invitational resource paper prepared for presentation at the International Association of Travel Behavior Research Conference to be held in Austin, Texas, September 1997.

Bhat, C.R. (1997b) An endogenous segmentation mode choice model with an application to intercity travel, Transportation Science, 31, 34-48.

Bhat, C.R. (1997c) A nested logit model with covariance heterogeneity, Transportation Research, 31B, 11-21.

Bhat, C.R. (1997d) Work mode choice and number of non-work commute stops, Transportation Research, 31B, 41-54.

Bhat, C.R. (1998a) Modeling the commute activity-travel pattern of workers: formulation and empirical analysis, Technical Paper, Department of Civil Engineering, University of Texas at Austin.

Bhat, C.R. (1998b) An analysis of travel mode and departure time choice for urban shopping trips, Transportation Research, 32B, 387-400.

Bhat, C.R. (1998c) Accommodating flexible substitution patterns in multidimensional choice modeling: formulation and application to travel mode and departure time choice, Transportation Research, 32B, 425-440.

Bhat, C.R. (1998d) Accommodating variations in responsiveness to level-of-service measures in travel mode choice modeling, Transportation Research, 32A, 495-507.

Bhat, C.R. (1998e) A post-home arrival model of activity participation behavior, Transportation Research, 32B, 361-371.

Bhat, C.R. and S.K. Singh (1999) A comprehensive daily activity-travel generation model system for workers, forthcoming, Transportation Research

Bhat, C. R., Guo, J. Y., Srinivasan, S., & Sivakumar, A. (2004).A comprehensive micro-simulator for daily activity-travel patterns. Proceedings of the conference on progress in activity-based models, Maastricht: EIRASS.

Bowman, J. L. (1995).Activity based travel demand model system with daily activity schedules(Master of Science thesis in transportation). Massachusetts Institute of Technology, Cambridge, MA.

Bowman, J. L. (1998).The day activity schedule approach to travel demand analysis(Ph.D. thesis). Massachusetts Institute of Technology, Cambridge, MA.

Bowman, J. L., & Ben-Akiva, M. (2000). Activity-based disaggregate travel demand model system with activity schedules.Transportation Research A,35, 1–28.

Bowman, J. L., Bradley, M., Shiftan, Y., Lawton, T. K., & Ben-Akiva, M. E. (1998).Demonstration of an activity-based model system for Portland. Proceedings 8th world conference on transport research, Antwerp.

Gärling, T., Brännäs, K., Garvill, J., Golledge, R. G., Gopal, S., Holm, E., & Lindberg, E. (1989). Household activity scheduling.Selected proceedings of the fifth world conference on transport research(Vol. 4, pp. 235–248). Ventura, CA: Western Periodicals.

Golledge, R. G., M.-P. Kwan and T. Gärling (1994) Computational-process modelling of house hold travel decisions using a geographical information system, Papers of the Regional Science Association, 73 (2) 99–117.

Hobeika, A. (2005) TRANSIMS overview, Technical Report, Virginia Polytechnic University, Virginia, July 2005, http://tmip.fhwa.dot.gov/resources/clearinghouse/docs/transims_fundamentals/ch1.pdf.

Jones, P. M., M. C. Dix, M. I. Clarke and I. G. Heggie (1983) Understanding Travel Behaviour, Gower, J. C., Aldershot.

Kitamura, R., & Fujii, S. (1998). Two computational process models of activity-travel choice. In T. Gärling, T. Laitila, & K. Westin (Eds.),Theoretical foundations of travel choice modelling (pp. 251–279). Oxford: Elsevier.

Miller, E. J., & Roorda, M. J. (2003). A prototype model of 24-hour scheduling for the Toronto area. Transportation Research Record,1831, 114–121.

Pendyala, R. M., Kitamura, R., Kikuchi,A.,Yamamoto, T., & Fujii, S. (2005). Florida activity mobility simulator, overview and preliminary validation results.Transportation Research Record, 1921, 123–130.

Recker, W. W., M. G. McNally and G. S. Root (1986a) A model of complex travel behavior: Part I - theoretical development, Transportation Research Part A: Policy and Practice, 20 (4) 307–318.

Recker, W. W., M. G. McNally and G. S. Root (1986b) A model of complex travel behavior: Part II - an operational model, Transportation Research Part A: Policy and Practice, 20 (4) 319–330.

Roorda, M. J. (2005).Activity-based modelling of household travel(PhD thesis). University of Toronto, Toronto.

Roorda, M. J., Doherty, S. T., & Miller, E. J. (2005). Operationalising household activity scheduling models, addressing assumptions and the use of new sources of behavioral data. In M. Lee-Gosselin & S. T. Doherty (Eds.),Integrated land-use and transportation models, behavioural foundations (pp. 61–85). Oxford: Elsevier.

Roorda, M. J., Miller, E. J., & Habib, K. M. N. (2008).Validation of TASHA:A 24-h activity scheduling microsimulation model.Transportation Research Part A,42, 360–375.

Roorda, M. J., & Miller, E. J. (2005). Strategies for resolving activity scheduling conflicts :An empirical analysis. In H. J. P. Timmermans (Ed.),Progress in activity-based analysis(pp. 203–222). Oxford: Elsevier.

Smith, LaRon; Beckman, Richard; Baggerly, Keith; Anson, Doug; Williams, Michael., TRANSIMS: TRansportation ANalysis and SIMulation System: Project Summary and Status, 1995, http://ntl.bts.gov/DOCS/466.html

- *Chapitre 2*

CHARYPAR, D. et al. (2008) Efficient algorithms for the microsimulation of travel behavior in very large scenarios (thesis for the degree of Doctor of Sciences), ETH Zurich.

CHARYPAR, D. et NAGEL, K. (2003) Generating Complete All-Day Activity Plans with Genetic Algorithms, Paper presented at the 10th International Conference on Travel Behaviour Research, Lucerne, August 10-14.

DAGNELIE, P. (1982) Diversité et unité de la statistique, journal de la société statistique de Paris, vol. 123, no 2, p. 86-92.

DENG, P. (1999) Genetic Algorithms with Multiple-Chromosome Crossover, Paper presented at the Network'99 Conference, Finland. http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.132.9713&rep=rep1&type=pdf

DUMAS, M. (1986) Discussion sur la définition du mot « statistique », journal de la société statistique de Paris, vol. 97, p. 253-258.

DURBIN, R., EDDY, S., KROGH, A., et MITCHISON, G. (1998) Biological sequence analysis « probabilistic models of proteins and nucleic acids », Cambridge University Press, Cambridge, UK.

DURET, L. (2011) Bioinformatique: Annotation des génomes (eucaryotes), support de cours, INSA Université Claude Bernard, Lyon.

GUEDJ, M. (2007) Méthodes Statistiques pour l'Analyse des Données Génétiques d'Association à Grande Échelle, Thèse pour le titre de Docteur en Statistique Génétique, Université d'Évry-Val d'Éssone.

JOH, C-H., ARENTZE, T.A., et TIMMERMANS, H.J.P. (1999) Multidimensional Sequence Alignment Methods for Activity Pattern Analysis: A comparison of dynamic programming and genetic algorithms, Paper prepared for presentation at the 39th ERSA Conference in Dublin, Ireland.

JONES, E.M., SHEEHAN, N.A., MASCA, N., WALLACE, S.E., MURTAGH, M.J. et BURTON, P.R. (2012) DataSHIELD – shared individual-level analysis without sharing the data: a biostatistical perspective, Norsk Epidemiologi no 21 (2), p.231-239

LABBE, A. (2013) De la Statistique à la Génétique : identifier les gènes responsables de maladies, Bulletin AMQ, Vol. LIII, no 2.

LE ROUX, T. (2014) Algorithme Génétique. Répéré le 12 juillet 2017 à http://igm.univ-mlv.fr/~dr/XPOSE2013/tleroux_genetic_algorithm/fonctionnement.html

MEISTER, K. et al. (2005) Generating daily activity schedules for households using Genetic Algorithms, paper presented at the 5th Swiss Transport Research Conference. DOI: 10.1007/s11116-005-5325-3

MOUNT, D.W. (2001) Bioiformatic « Sequence and Genome Analysis », Cold Sring Harbor Laboratory Press, New York, USA.

MOURAD, R., SINOQUET, C., et LERAY, P. (2010) Probabilistic graphical models for genetic association studies, Briefings in bioinformatics, vol 13. No 1, p. 20-33. doi:10.1093/bib/bbr015

MUKHOPADHYAY, D.M., BALITANAS, M.O., ALISHEROV, F.A., JEON ,S.-H. et BHATTACHARYYA, D. (2009) Genetic Algorithm: A Tutorial Review, International Journal of Grid and Distributed Computing, Vol.2, No.3, September, 2009.

RENYES, C. (2007) Etude des Algorithmes génétiques et application aux données de protéomique. Sciences du Vivant [q-bio]. Repéré le 12 juillet 2016 à https://tel.archives-ouvertes.fr/tel-00268927/document

RUMELHARD, G. (2006). Analyse statistique de l'ADN, modélisation à l'aide des chaînes de Markov, simulation et détection de biais, Biologie-Géologie (APBG) n°6. www.mathom.fr/mathom/sauvageot/Modelisation/ADN/ADN-Markov.pdf

SAADI, I. , MUSTAFA, A., TELLER, J., et COOLS, M. (2016a) An integrated framework for forecasting travel behavior using Markov Chain Monte-Carlo simulation and profile Hidden Markov Models, Proceedings of the 95th Annual Meeting of the Transportation Research Board.

SAADI, I., MUSTAFA, A., TELLER, J., FAROOQ, B., et COOLS, M. (2016b) Hidden Markov Model-based population synthesis, Transportation Research Part B: Methodological vol. 90 p.1-21, Pergamon.

SAMMOUR, G., BELLEMANS, T., VANHOOF, K., JANSSENS, D., et WETS, G. (2012) The usefulness of the sequence alignement methods in validating rule-based activity-based models, TRB 2012 Annual Meeting.

SHOVAL, N., et ISAACSON, M. (2007) Sequence Alignment as a Method for Human Activity Analysis in Space and Time, *Annals of the Association of American Geographers*, 97(2), pp. 282–297, Blackwell Publishing, Oxford.

SILLANPAA, M.J. (2011) Overview of techniques to account for confounding due to population stratification and cryptic relatedness in genomic data association analyses, Heredity vol. 106, p.511–519, Macmillan Publishers

TORRENS-INERN, J. (1956) Variété. Qu'est-ce que la statistique?, journal de la société statistique de Paris, vol. 97, p. 289-296.

WHITLEY, D. (1994) A Genetic Algorithm Tutorial, Statistics and Computing, vol.4, p. 65-85.

WILSON, W.C. (1998) Activity pattern analysis by means of sequence-alignment methods, Environment and Planning, volume 30, p. 1017-1038.

TOLLARI, S. (2003) Algorithmes génétiques. Répéré le 12 juillet 2017 à http://sis.univ-tln.fr/~tollari/TER/AlgoGen1/node5.html

Biostatistique. (s. d.). Dans Wikipédia, l'encyclopédie libre. Repéré le 5 décembre 2016 à https://fr.wikipedia.org/wiki/Biostatistique

Biologie. (s. d.). Dans Wikipédia, l'encyclopédie libre. Repéré le 5 décembre 2016 à https://fr.wikipedia.org/wiki/Biologie

Statistique. (s. d.). Dans Wikipédia, l'encyclopédie libre. Repéré le 5 décembre 2016 à https://fr.wikipedia.org/wiki/Statistique

Statistique. (s. d.). Dans Larousse, Le dictionnaire en ligne. Repéré le 5 décembre 2016 à http://www.larousse.fr/dictionnaires/francais/statistique/74516

Mathematical Optimization. (s. d.). Dans Wikipédia, l'encyclopédie libre. Repéré le 12 juillet 2016 à https://en.wikipedia.org/wiki/Mathematical_optimization

Evolutionary Algorithm. (s. d.). Dans Wikipédia, l'encyclopédie libre. Repéré le 12 juillet 2016 à https://en.wikipedia.org/wiki/Evolutionary_algorithm

Algorithme Génétique. (s. d.). Dans Wikipédia, l'encyclopédie libre. Repéré le 12 juillet 2016 à https://fr.wikipedia.org/wiki/Algorithme_génétique

Génomique. (s. d.). Dans Wikipédia, l'encyclopédie libre. Repéré le 12 juillet 2016 à https://fr.wikipedia.org/wiki/G%C3%A9nomique

Méthode de Monte-Carlo par chaînes de Markov. (s. d.). Dans Wikipédia, l'encyclopédie libre. Repéré le 12 juillet 2016 à https://fr.wikipedia.org/wiki/M%C3%A9thode_de_Monte-Carlo_par_cha%C3%AEnes_de_Markov

Chaînes de Markov. (s. d.). Dans Wikipédia, l'encyclopédie libre. Repéré le 12 juillet 2016 à https://fr.wikipedia.org/wiki/Cha%C3%AEne_de_Markov

Modèle à Markov Caché. (s. d.). Dans Wikipédia, l'encyclopédie libre. Repéré le 12 juillet 201de à https://fr.wikipedia.org/wiki/Mod%C3%A8le_de_Markov_cach%C3%A9

- *Chapitre 3*

Introductory Course in Statistics using R (Support de cours) de M. COOLS.

TFE - Utilisation de modèles et techniques biostatistiques pour la modélisation des phénomènes de transport

Pierre Cuenca
Master ingénieur-architecte,
Université de Liège

- Annexe 1 -

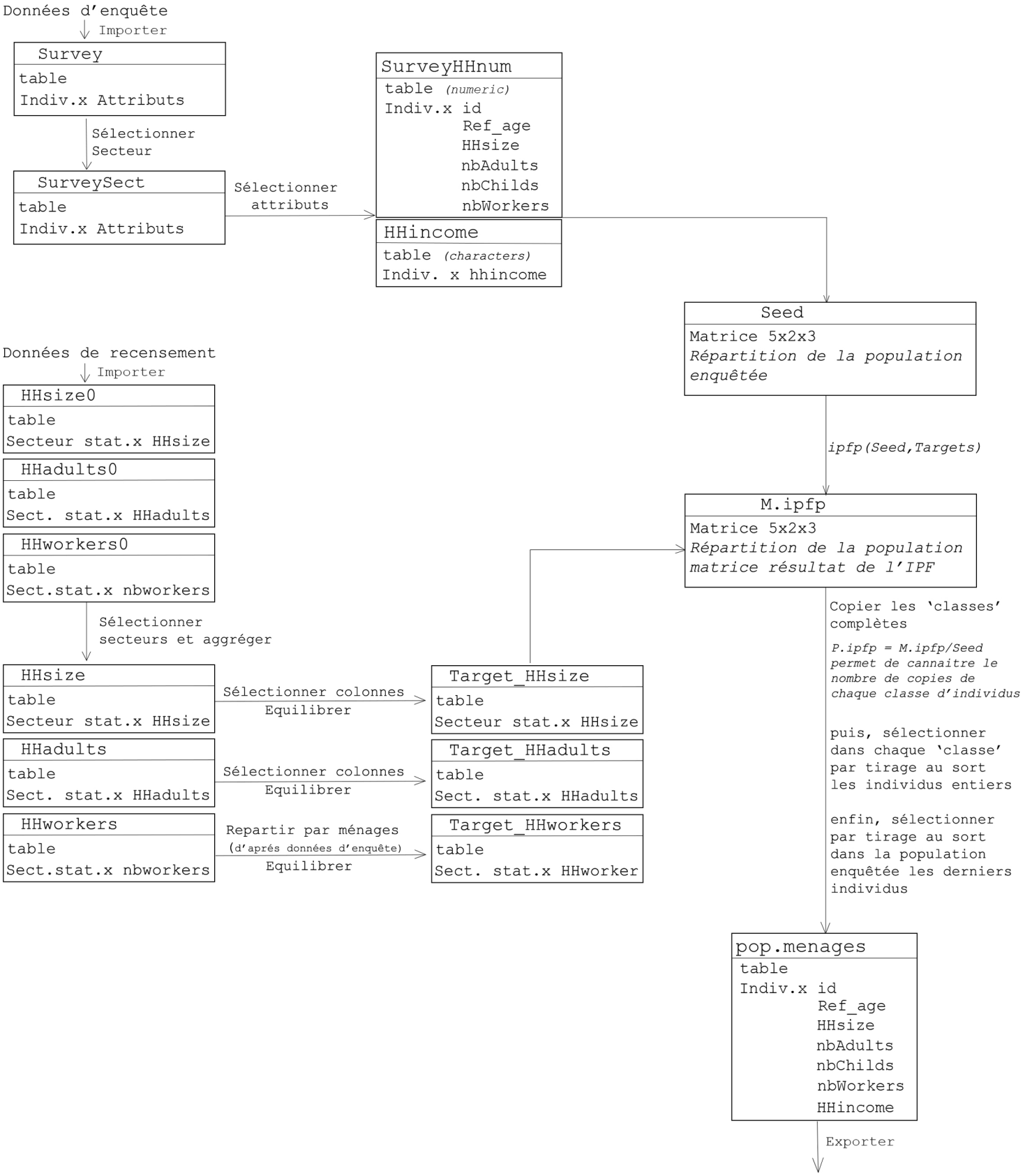
Codes en langage R des 4 modules développés

Organisation de l'annexe

- Module 1. (4+1 pages)
- Module 2. (4+1 pages)
- Module 3.1. (3+1 pages)
- Module 3.2. (6+1pages
- Module 4. (5+1 pages)
- Fonctions (7 pages)

Août 2017

MODULE 1. CREATION D'UNE POP. SYNTH. DE MENAGES



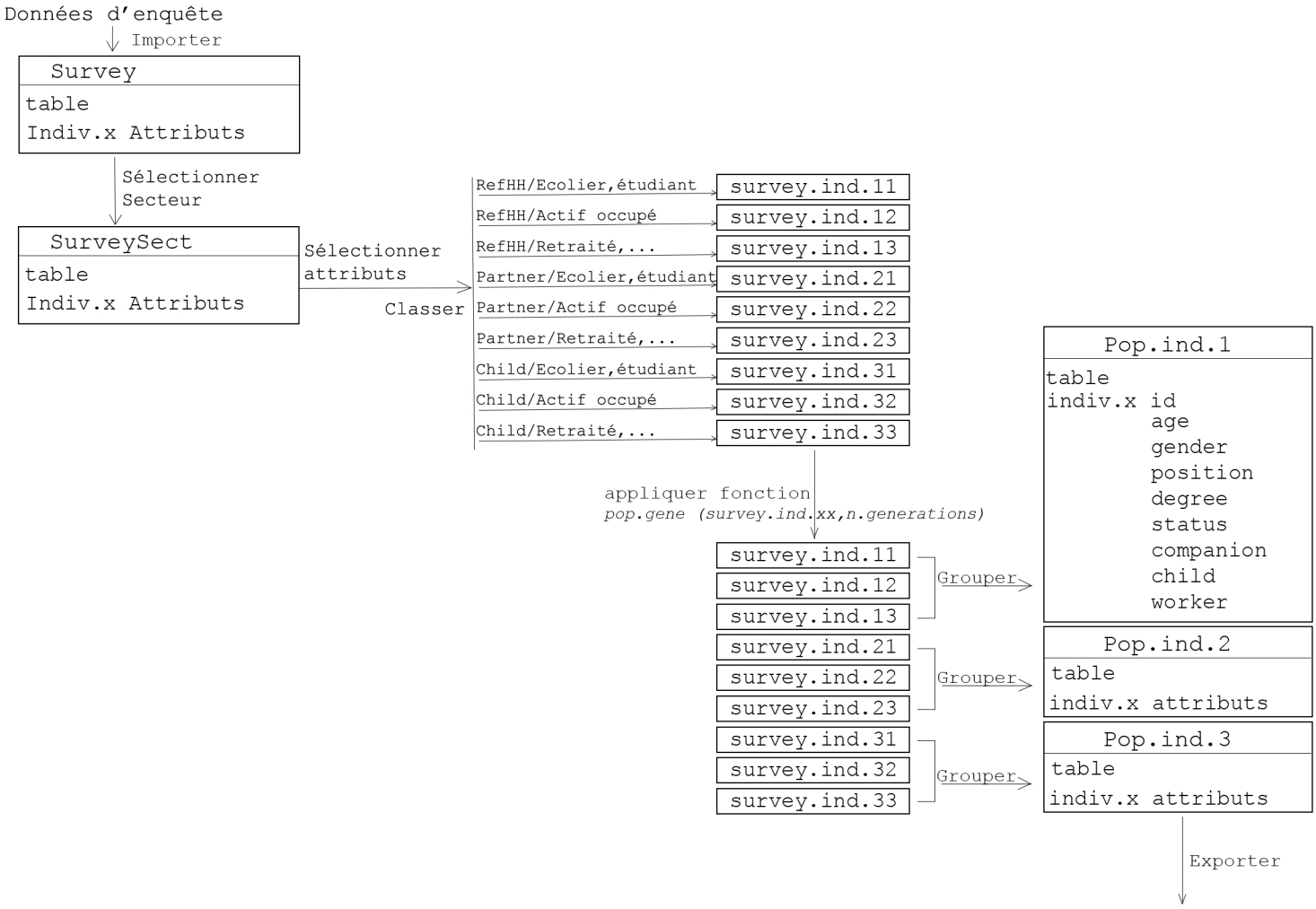
```
#Module 1. CREATION DE POP. SYNTHETIQUE DE MENAGES selon une méthode IPF
#
#Auteur : Pierre CUENCA
#20.07.2017
#-----------------------------------------------------------------------------------------------------------
# Création des foyers
# Attributs : Age du chef de famille/nbindiv/nbadults/nbchilds/nbworkers/HHincomepworker
#-----------------------------------------------------------------------------------------------------------
#Specification du chemin du dossier
Chemin.Data <- "D:/_Travail/_TFE/_R.Bases de données/"

#Chargement des fonctions annexes appelées dans le programme
source('D:/_Travail/_TFE/_R.CodeTFE/traitement.donnee.census.R')
source('D:/_Travail/_TFE/_R.CodeTFE/Ipfp.R')
source('D:/_Travail/_TFE/_R.CodeTFE/Ipfp2.R')

#Importation des données d'entrée'
load(paste(Chemin.Data,"pierre.rda",sep=""))

#the following code select the complete cases (removes the cases with missing values)
Survey <- pierre.dat[complete.cases(pierre.dat),]

#############################################################################################################
#### Sélection d'un secteur géographique

Secteur <- 62063    # Ville de Liège (code des secteurs statistiques)

SurveySect <- NULL

for (i in 1:nrow(Survey)){
  if (is.na(Survey[i,27]) == FALSE){     # s'il ya des valeurs NA dans le tableau
    if (Survey[i,27] == Secteur)
    {SurveySect <- rbind(SurveySect,Survey[i,])
    }
  }
}

#############################################################################################################
#### Création d'un tableau pour les données associées aux ménages
# On sélectionne les attributs qui nous intéressent
# on isole chaque foyer en identifiants le "reference person HH"

#création du tableau de donnée en selectionnant les attributs de tous les individus de référence

HHIncome <- NULL
SurveyHHnum <- NULL

for (i in 1:nrow(SurveySect)){
   if (SurveySect[i,7] == "Reference person HH"){
       HHIncome <-c(HHIncome,as.character(Survey[i,36]))
       SurveyHHnum <- rbind(SurveyHHnum,
c(SurveySect[i,5],SurveySect[i,37],SurveySect[i,39],SurveySect[i,38],SurveySect[i,40]))
   }
}

colnames(SurveyHHnum) <- c("Ref_Age","HHsize","NbAdults","NbChilds","Nbworkers")
SurveyHH<-cbind(SurveyHHnum,HHIncome)

#############################################################################################################
# Récupération données de recensement
# Importation des données depuis des tableaux excel (enregistré au format .csv (avec séparation ";"))

HHsize0 <- read.table(paste(Chemin.Data,"Pour
export/TailleHH2.csv",sep=""),sep=";",header=TRUE,stringsAsFactors=FALSE)
Nbadults0 <- read.table(paste(Chemin.Data,"Pour
export/Couples_famille.csv",sep=""),sep=";",header=TRUE,stringsAsFactors=FALSE)
Nbworkers0 <- read.table(paste(Chemin.Data,"Pour export/Actifs
Occupés2.csv",sep=""),sep=";",header=TRUE,stringsAsFactors=FALSE)
```

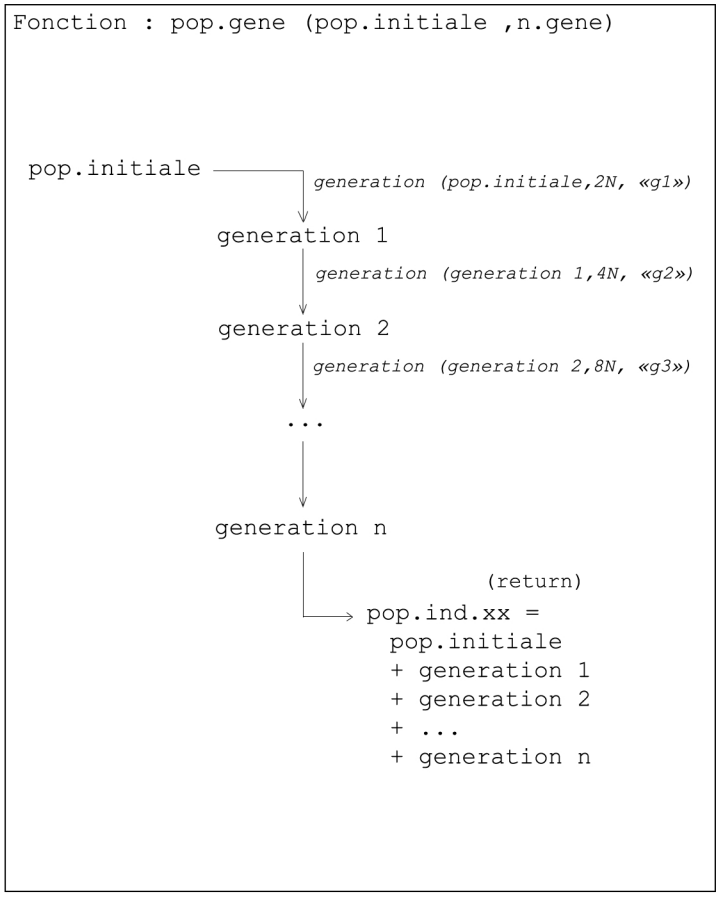
```
# Extraction des données associées au secteur statistique modélisé

SecteurId <- c("62063A00-","62063A01-","62063A02-") # il a d'autres secteurs !

#Chargement de la fonction traitement.donnee.census
#cette fonction permet de sélectionner dans les tables les secteurs de la zone d'étude
#et aggrége les données dans un vecteur de données

#####   Taille des ménages (>HHsize)
# HHsize est le tableau qui reprends les valeurs pour le Secteur global
HHsize <- traitement.donnee.census(HHsize0,SecteurId)

#### Nombre d'adulte dans le ménage (>Nbadults)
Nbadults <- traitement.donnee.census(Nbadults0,SecteurId)

#### Nombre de travailleurs dans le ménage (>Nbworkers)
Nbworkers <- traitement.donnee.census(Nbworkers0,SecteurId)

###############################################################################################################
#Création des ménages synthétiques en utilisant la méthode IPF

#creation d'un tableau de dimension 3 (nb de "contraintes") pour croiser les données de l'échantion de
l'enquête
#Remarque : c'est un comptage

L1 <- max(SurveyHHnum[,2])  #dimension 1 Taille du ménage
L2 <- max(SurveyHHnum[,3])  #dimension 2 Nombre d'adultes dans le ménage
L3 <- max(SurveyHHnum[,5])+1  #dimension 3 Nombre de travailleurs dans le ménage

Seed <- array(0, dim=c(L1,L2,L3))

for (i in 1:L1){
  for (j in 1:L2){
    for (k in 1:L3){
      for (n in 1:nrow(SurveyHHnum)){
        if (SurveyHHnum[n,2] == i & SurveyHHnum[n,3] == j & SurveyHHnum[n,5] == k-1){
          Seed[i,j,k] <- Seed[i,j,k]+1
        }
      }
    }
  }
}

#pour la donnée du nombre de travailleur, on ne dispose que du total de tavailleur dans le secteur
#on ne connait pas la confoguration des méganes (nbtravailleur/ménage)
#on va donc recréer la donnée en se basant sur les données d'enquête et en ramenant le total de travaileurs
#à la valeur cible fournie par le recensement

Target_workers <- c(sum(Seed[,,1]),sum(Seed[,,2]),sum(Seed[,,3]))

s_surv <- Target_workers[2]*1 + Target_workers[3]*2    #Nb total de travailleurs (enquête)
s_targ <- Nbworkers[length(Nbworkers)]                 #Nb total de travailleurs (census)

Target_workers <- Target_workers*(s_targ/s_surv)

# créons aussi des tableaux de valeurs cibles pour le taille des ménages et le nombre d'adulte par ménage

Target_size <- HHsize[1]
for (i in 2:L1){
  Target_size <- c(Target_size, HHsize[i])
}

Target_adults <- c(Nbadults[7]+HHsize[1],Nbadults[4])

# "Balancing" > equilibrage du nombre de ménages

s_size <- sum(Target_size)          #nb de ménages pris en compte pour la taille des ménages
s_adults <- sum(Target_adults)      #nb de ménages pris en compte pour le nombre d'adults par ménage
```

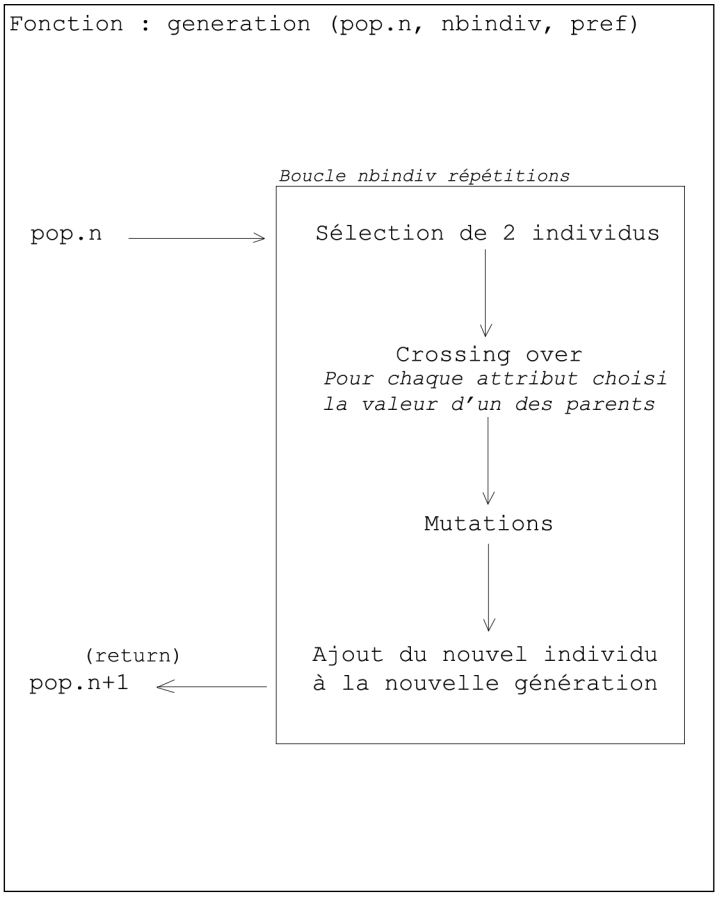
```
s_workers <- sum (Target_workers)   #nb de ménages pris en compte pour le nombre de travailleurs par ménage

#la valeur s_workers n'est pas une valeur de référence
#c'est une caractéristique de l'échantillon d'étude et non de la population réelle totale
#Prenons comme valeur cible une moyenne du nombre de ménages issus des données de taille et d'adultes

S <- (s_size+s_adults)/2

Target_size <- Target_size*(S/s_size)
Target_adults <- Target_adults*(S/s_adults)
Target_workers <- Target_workers*(S/s_workers)

#IPF
target.list <- list(1,2,3)
target.data <- list(Target_size,Target_adults,Target_workers)

M.ipfp <- ipfp(Seed, target.list, target.data, 100, 1e-10)

Seed.div <- Seed
Seed.div[Seed.div == 0] <- 1
P.ipfp <- M.ipfp/Seed.div # crée une matrice de 'proportions'

#IPFP existant
#source('D:/_Travail/_TFE/_R.CodeTFE/ipfp_multi_dim.R')
#Ipfp(Seed,target.list, target.data, print = FALSE, iter = 1000, tol = 1e-10, tol.margins = 1e-10, na.target
= FALSE)

#############################################################################################################
# Utilisation des proportions fournies par la procédure IPF pour créer la population synthtique de ménages

P.ipfp[P.ipfp<1e-3]<-0

#copie des individu "entiers"
P.ipfp.t <- trunc(P.ipfp,0)

i <- 1
id <- 1
Population <- NULL

for (i in 1:nrow(SurveyHHnum)){
u <- P.ipfp.t[SurveyHHnum[i,2],SurveyHHnum[i,3],SurveyHHnum[i,5]+1]    #On se sert des valeurs comme index
if(u > 0){
  for (n in 1:u){
      Population <- rbind(Population,c(id,SurveyHH[i,]))
      id<-id+1
    }
  }
}

#tirage au sort des individus par "classe" de contrainte
#Pour chacune des classes
#Nr = nombre d'individu a ajouter pour chaque classe (matrice multidim)

P.ipfp.r <- P.ipfp-P.ipfp.t
Nr <- trunc(Seed*P.ipfp.r)

for(i in 1:L1){
  for(j in 1:L2){
   for(k in 1:L3){
        #s'il y a des individus qui doivent être sélectionnés
     if (Nr[i,j,k] > 0){
        #Création d'une liste de candidat (vecteur de propabilité de tirage 0 ou 1)
       if (SurveyHHnum[1,2] == i & SurveyHHnum[1,3] == j & SurveyHHnum[1,5] == k-1){
         Cand <- 1
         } else{
             Cand <- 0
         }
         for (n in 2:nrow(SurveyHHnum)){
           if(SurveyHHnum[n,2] == i & SurveyHHnum[n,3] == j & SurveyHHnum[n,5] == k-1){
```

```
            Cand <- c(Cand,1)
          } else{
            Cand <- c(Cand,0)
          }
        }
        #Sélection aléatoire sans remise
        samp <- sample(c(1:nrow(SurveyHHnum)),Nr[i,j,k],replace=FALSE,Cand)
        # ajout des candidats à la population synthétique
        for (m in 1:length(samp)){
            Population <- rbind(Population, c(id,SurveyHH[samp[m],]))
            id <-id+1
        }
    }
   }
  }
}
  #Tirage au sort selon proportion restantes dans toute la population

R <- S-nrow(Population) #nombre de personnes à tirer au sort
Pr <- (Seed*P.ipfp.r)-Nr #probablilité d'être tiré au sort selon la classe

if(R>0){

  #Creation liste de probabilite de chaque individu

  Prob <- Pr[SurveyHHnum[1,2],SurveyHHnum[1,3],SurveyHHnum[1,5]+1]

  for (i in 2:nrow(SurveyHHnum)){
    Prob <- c(Prob,Pr[SurveyHHnum[i,2],SurveyHHnum[i,3],SurveyHHnum[i,5]+1])
  }

  #Tirage avec remise
  #Si on effectue une tirage sans remise, il faut réévaluer les prbabilité à chaque itération
  samp <- sample(c(1:nrow(SurveyHHnum)),R,replace=TRUE,Prob)
  #ajout des candidats sélectionnés à la population
  for (m in 1:length(samp)){
      Population <- rbind(Population, c(id,SurveyHH[samp[m],]))
      id <-id+1
  }
}

#########################################################################################################
#Vérification pop synth / pop réelle
Ecart <- 0
i=5
for (i in 1:4){

Ecart<-max(Ecart,abs((length(Population[Population[,"HHsize"]==i,"HHsize"])-Target_size[i])/Target_size[i]))
}
for (i in 1:2){

Ecart<-max(Ecart,abs((length(Population[Population[,"NbAdults"]==i,"NbAdults"])-Target_adults[i])/Target_adul
ts[i]))
}
for (i in 1:3){

Ecart<-max(Ecart,abs((length(as.numeric(Population[Population[,"Nbworkers"]==i-1,"Nbworkers"]))-Target_worker
s[i])/Target_workers[i]))
}
print(Ecart)
#########################################################################################################
#Exportation de la table Population
colnames(Population) <- c("id","Ref_Age","HHsize","NbAdults","NbChilds","Nbworkers","HHincome")
write.table(Population,file=paste(Chemin.Data,"Save table/Pop.menages.csv",sep=""),append=FALSE,
sep=";",row.names=FALSE,col.names=TRUE)
remove(list=ls())
#########################################################################################################
#fin du module 1.
#########################################################################################################
```

MODULE 2. ETENDRE LA VARIABILITE DES INDIVIDUS D'UNE POP. SYNTHETIQUE







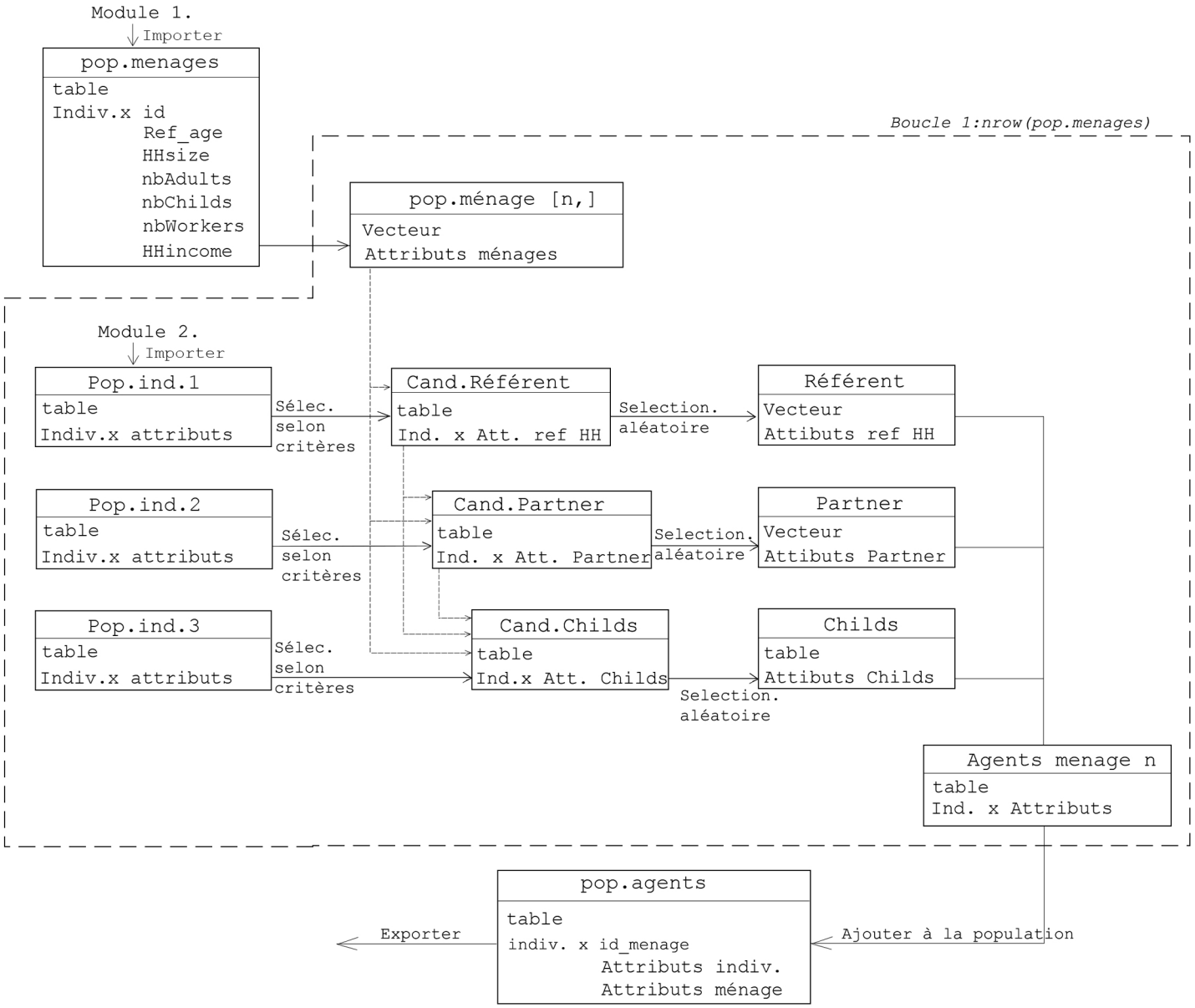
```
#Module 2. ETENDRE LA VARIALIBILTE DES INDIVIDUS D'UNE POPULATION SYNTHETIQUE
#
#Auteur : Pierre CUENCA
#25/07/2017
#-----------------------------------------------------------------------------------------------------------
#Production d'une grande diversité d'individus à partir des données d'enquête
#Chaque nouvelle génération à le double de population par rapport à la popuation qui l'a engendrée
#-----------------------------------------------------------------------------------------------------------
#Specification du chemin du dossier
Chemin.Data <- "D:/_Travail/_TFE/_R.Bases de données/"
#Chargement des fonctions annexes appelées par le programme
source('D:/_Travail/_TFE/_R.CodeTFE/pop.gene.R')
source('D:/_Travail/_TFE/_R.CodeTFE/Generation.R')
source('D:/_Travail/_TFE/_R.CodeTFE/Generation.student.R')
source('D:/_Travail/_TFE/_R.CodeTFE/pop.gene.student.R')

### Importation des données - Enquete

#chargement du tableau de donnée complet
load(paste(Chemin.Data,"pierre.rda",sep=""))

#Sélection des cases avec des données complétes
Survey <- pierre.dat[complete.cases(pierre.dat),]

#############################################################################################################
####Sélection des données pour le secteur géographique

Secteur <- 62063    # Ville de Liège (code des secteurs statistiques)

  SurveySect <- NULL

for (i in 1:nrow(Survey)){
  if (is.na(Survey[i,27]) == FALSE){     # il ya des valeurs NA dans le tableau
    if (Survey[i,27] == Secteur)
    {SurveySect <- rbind(SurveySect,Survey[i,])
      }
    }
  }
#############################################################################################################
####Création d'une table pour les données associées aux individus
# Sélection des attributs utiles pour notre modèle
# répartition des individus en 3x3 = 9 classes
# le premier critère d'appartenance à une classe est la position dans le ménage
# Reference person HH (1) / Partner (2) / Child (3)
# le deuxième critère est le status
# Etudiant ou écolier (1) / actif occupé (2) / Retraité, chercheur d'emploi ou personne au foyer (3)

#Initialisation : création des vecteurs

Survey.ind.11 <- NULL
Survey.ind.12 <- NULL
Survey.ind.13 <- NULL
Survey.ind.21 <- NULL
Survey.ind.22 <- NULL
Survey.ind.23 <- NULL
Survey.ind.31 <- NULL
Survey.ind.32 <- NULL
Survey.ind.33 <- NULL

#Table de données d'enquête

for (n in 1:nrow(SurveySect)){
  if (SurveySect[n,"position"] == "Reference person HH"){
    if (SurveySect[n,"status"] == "Pupil, student"){
      Survey.ind.11 <- rbind(Survey.ind.11, c(SurveySect[n,"i_id"], SurveySect[n,"age"],
as.character(SurveySect[n,"gender"]), as.character(SurveySect[n,"position"]),
as.character(SurveySect[n,"degree"]), as.character(SurveySect[n,"status"]),
as.character(SurveySect[n,"companion"]), as.character(SurveySect[n,"child"])))
```

```
    }
    else if (SurveySect[n,"status"] == "(Pre)retired" | SurveySect[n,"status"] == "Housewife/househusband" |
SurveySect[n,"status"] == "Unemployed" | SurveySect[n,"status"]=="Incapacitated"){
      Survey.ind.13 <- rbind(Survey.ind.13, c(SurveySect[n,"i_id"], SurveySect[n,"age"],
as.character(SurveySect[n,"gender"]), as.character(SurveySect[n,"position"]),
as.character(SurveySect[n,"degree"]), as.character(SurveySect[n,"status"]),
as.character(SurveySect[n,"companion"]), as.character(SurveySect[n,"child"])))
    }
    else {
      Survey.ind.12 <- rbind(Survey.ind.12, c(SurveySect[n,"i_id"], SurveySect[n,"age"],
as.character(SurveySect[n,"gender"]), as.character(SurveySect[n,"position"]),
as.character(SurveySect[n,"degree"]), as.character(SurveySect[n,"status"]),
as.character(SurveySect[n,"companion"]), as.character(SurveySect[n,"child"])))
    }
  }
  else if (SurveySect[n,"position"] == "Partner"){
    if (SurveySect[n,"status"] == "Pupil, student"){
      Survey.ind.21 <- rbind(Survey.ind.21, c(SurveySect[n,"i_id"], SurveySect[n,"age"],
as.character(SurveySect[n,"gender"]), as.character(SurveySect[n,"position"]),
as.character(SurveySect[n,"degree"]), as.character(SurveySect[n,"status"]),
as.character(SurveySect[n,"companion"]), as.character(SurveySect[n,"child"])))
    }
    else if (SurveySect[n,"status"] == "(Pre)retired" | SurveySect[n,"status"] == "Housewife/househusband" |
SurveySect[n,"status"] == "Unemployed" | SurveySect[n,"status"]=="Incapacitated" ){
      Survey.ind.23 <- rbind(Survey.ind.23, c(SurveySect[n,"i_id"], SurveySect[n,"age"],
as.character(SurveySect[n,"gender"]), as.character(SurveySect[n,"position"]),
as.character(SurveySect[n,"degree"]), as.character(SurveySect[n,"status"]),
as.character(SurveySect[n,"companion"]), as.character(SurveySect[n,"child"])))
    }
    else {
      Survey.ind.22 <- rbind(Survey.ind.22, c(SurveySect[n,"i_id"], SurveySect[n,"age"],
as.character(SurveySect[n,"gender"]), as.character(SurveySect[n,"position"]),
as.character(SurveySect[n,"degree"]), as.character(SurveySect[n,"status"]),
as.character(SurveySect[n,"companion"]), as.character(SurveySect[n,"child"])))
    }
}
  else {
    if (SurveySect[n,"status"] == "Pupil, student"){
      Survey.ind.31 <- rbind(Survey.ind.31, c(SurveySect[n,"i_id"], SurveySect[n,"age"],
as.character(SurveySect[n,"gender"]), as.character(SurveySect[n,"position"]),
as.character(SurveySect[n,"degree"]), as.character(SurveySect[n,"status"]),
as.character(SurveySect[n,"companion"]), as.character(SurveySect[n,"child"])))
    }
    else if (SurveySect[n,"status"] == "(Pre)retired" | SurveySect[n,"status"] == "Housewife/househusband" |
SurveySect[n,"status"] == "Unemployed" | SurveySect[n,"status"]=="Incapacitated"){
      Survey.ind.33 <- rbind(Survey.ind.33, c(SurveySect[n,"i_id"], SurveySect[n,"age"],
as.character(SurveySect[n,"gender"]), as.character(SurveySect[n,"position"]),
as.character(SurveySect[n,"degree"]), as.character(SurveySect[n,"status"]),
as.character(SurveySect[n,"companion"]), as.character(SurveySect[n,"child"])))
    }
    else {
      Survey.ind.32 <- rbind(Survey.ind.32, c(SurveySect[n,"i_id"], SurveySect[n,"age"],
as.character(SurveySect[n,"gender"]), as.character(SurveySect[n,"position"]),
as.character(SurveySect[n,"degree"]), as.character(SurveySect[n,"status"]),
as.character(SurveySect[n,"companion"]), as.character(SurveySect[n,"child"])))
    }
  }
}


#Noms de colonnes
colnames(Survey.ind.11)<-c("i_id","age","gender","position","degree","status","companion","child")
colnames(Survey.ind.12)<-c("i_id","age","gender","position","degree","status","companion","child")
colnames(Survey.ind.13)<-c("i_id","age","gender","position","degree","status","companion","child")
colnames(Survey.ind.21)<-c("i_id","age","gender","position","degree","status","companion","child")
colnames(Survey.ind.22)<-c("i_id","age","gender","position","degree","status","companion","child")
colnames(Survey.ind.23)<-c("i_id","age","gender","position","degree","status","companion","child")
colnames(Survey.ind.31)<-c("i_id","age","gender","position","degree","status","companion","child")
colnames(Survey.ind.32)<-c("i_id","age","gender","position","degree","status","companion","child")
```

```
colnames(Survey.ind.33)<-c("i_id","age","gender","position","degree","status","companion","child")

# Remarque : on peut créer une classe de plus
# et il faudrait différencier Pre-retired et retired et puis pupil et Student!
#########################################################################################################
####Expension des groupes d'individus en ultilisant des techniques de croisement et mutations

pop.ind.11 <- pop.gene.student (Survey.ind.11,5) # On fait apparaitre une corrélation entre age et degree
dans les classes pupil/student
pop.ind.12 <- pop.gene (Survey.ind.12,5)

# D <- pop.ind.12[!(duplicated(pop.ind.12[,-1])),]#on enleve les doublons
# nrow(Survey.ind.12)           #taille pop initiale de la classe
# nrow(pop.ind.12)             #taille pop générée
# nrow(D)                      #taille pop sans doublons

pop.ind.13 <- pop.gene (Survey.ind.13,5)
pop.ind.21 <- pop.gene.student (Survey.ind.21,5)
pop.ind.22 <- pop.gene (Survey.ind.22, 5)
pop.ind.23 <- pop.gene (Survey.ind.23, 5)
pop.ind.31 <- pop.gene.student (Survey.ind.31,5)
pop.ind.32 <- pop.gene (Survey.ind.32, 5)
pop.ind.33 <- pop.gene (Survey.ind.33, 5)

rownames(pop.ind.11)<-NULL
rownames(pop.ind.12)<-NULL
rownames(pop.ind.13)<-NULL
rownames(pop.ind.21)<-NULL
rownames(pop.ind.22)<-NULL
rownames(pop.ind.23)<-NULL
rownames(pop.ind.31)<-NULL
rownames(pop.ind.32)<-NULL
rownames(pop.ind.33)<-NULL

#########################################################################################################
#Pour la suite, il n'est plus nécessaire de considérer les neufs classes, on peut ne conserver que 3 classes
#1. reference personn HH 2. Partner 3. Child
#Ces classes seront utiles pour "regrouper" les individus en foyers

#On ajoute une ligne "worker" pour différencier les actifs ayant du travail
T1<-cbind(pop.ind.11,rep("No",nrow(pop.ind.11)),stringsAsFactors=FALSE)
colnames(T1)<-c(colnames(pop.ind.11),"worker")
T2<-cbind(pop.ind.12,rep("Yes",nrow(pop.ind.12)),stringsAsFactors=FALSE)
colnames(T2)<-c(colnames(pop.ind.12),"worker")
T3<-cbind(pop.ind.13,rep("No",nrow(pop.ind.13)),stringsAsFactors=FALSE)
colnames(T3)<-c(colnames(pop.ind.13),"worker")
pop.ind.1 <- rbind (T1,T2,T3)

T1<-cbind(pop.ind.21,rep("No",nrow(pop.ind.21)),stringsAsFactors=FALSE)
colnames(T1)<-c(colnames(pop.ind.21),"worker")
T2<-cbind(pop.ind.22,rep("Yes",nrow(pop.ind.22)),stringsAsFactors=FALSE)
colnames(T2)<-c(colnames(pop.ind.22),"worker")
T3<-cbind(pop.ind.23,rep("No",nrow(pop.ind.23)),stringsAsFactors=FALSE)
colnames(T3)<-c(colnames(pop.ind.23),"worker")
pop.ind.2 <- rbind (T1,T2,T3)

T1<-cbind(pop.ind.31,rep("No",nrow(pop.ind.31)),stringsAsFactors=FALSE)
colnames(T1)<-c(colnames(pop.ind.31),"worker")
T2<-cbind(pop.ind.32,rep("Yes",nrow(pop.ind.32)),stringsAsFactors=FALSE)
colnames(T2)<-c(colnames(pop.ind.32),"worker")
T3<-cbind(pop.ind.33,rep("No",nrow(pop.ind.33)),stringsAsFactors=FALSE)
colnames(T3)<-c(colnames(pop.ind.33),"worker")
pop.ind.3 <- rbind (T1,T2,T3)

rm(T1,T2,T3)
rm(pop.ind.11,pop.ind.12,pop.ind.13,pop.ind.21,pop.ind.22,pop.ind.23,pop.ind.31,pop.ind.32,pop.ind.33)

#########################################################################################################
#L'age minimum des enfants dans la table pop.ind.3 est de 7 ans
```

```
#On va étendre la tables
#Les attributs autres que 'gender' sont identiques pour les enfants de moins de 12 ans

      pop.ind.3.ext <- pop.ind.3[pop.ind.3[,"age"] < 12,]
      pop.ind.3.ext[,"age"] <- pop.ind.3.ext[,"age"]-6
      pop.ind.3.ext[,"i_id"] <- "xxx"

      pop.ind.3 <- rbind(pop.ind.3,pop.ind.3.ext)
rm(pop.ind.3.ext)

#############################################################################################################
#Exportation des tables

write.table(pop.ind.1,file=paste(Chemin.Data,"Save table/pop.ind.1.csv",sep=""),append=FALSE,
sep=";",row.names=FALSE,col.names=TRUE)
write.table(pop.ind.2,file=paste(Chemin.Data,"Save table/pop.ind.2.csv",sep=""),append=FALSE,
sep=";",row.names=FALSE,col.names=TRUE)
write.table(pop.ind.3,file=paste(Chemin.Data,"Save table/pop.ind.3.csv",sep=""),append=FALSE,
sep=";",row.names=FALSE,col.names=TRUE)
remove(list=ls())

#############################################################################################################
#Fin du Module 2.
#############################################################################################################
```

MODULE 3.1. CREATION D'UNE POP. SYNTHETIQUE D'INDIVIDUS : Remplissage des foyers

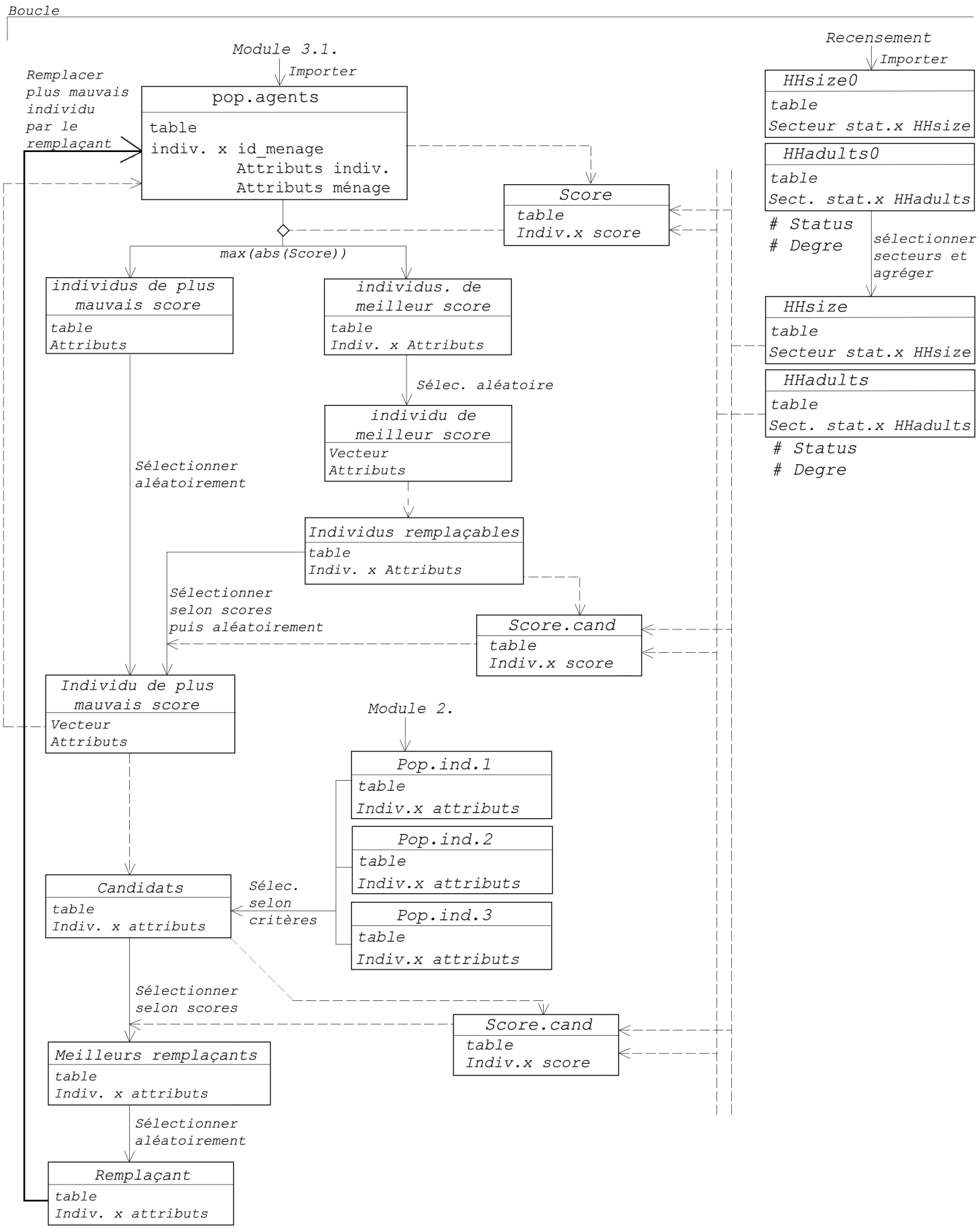


```
#Module 3.1. CREATION D'UNE POPULATION SYNTHETIQUE D'INDIVIDU - remplissage des foyers
#
#Auteur : Pierre CUENCA
#26.07.2017
#-----------------------------------------------------------------------------------------------------------
# Attribue des agents à chaque ménage de la population synthetique de ménages créée dans le module 1.
# Cela permet de créer une population 'probable' d'agents
# On vérifie la cohérence des ages, genre, position, status,etc... au sein des ménages
# MAIS cette population ne vérifie pas forcément les caractéristiques des données de recensement
#-----------------------------------------------------------------------------------------------------------
#Récupération des données des modules 1 et 2
#Specification du chemin du dossier

Chemin.Data <- "D:/_Travail/_TFE/_R.Bases de données/"

pop.menages <- read.table(paste(Chemin.Data,"Save
table/Pop.menages.csv",sep=""),sep=";",header=TRUE,stringsAsFactors=FALSE)
pop.ind.1<- read.table(paste(Chemin.Data,"Save
table/pop.ind.1.csv",sep=""),sep=";",header=TRUE,stringsAsFactors=FALSE)
pop.ind.2<- read.table(paste(Chemin.Data,"Save
table/pop.ind.2.csv",sep=""),sep=";",header=TRUE,stringsAsFactors=FALSE)
pop.ind.3<- read.table(paste(Chemin.Data,"Save
table/pop.ind.3.csv",sep=""),sep=";",header=TRUE,stringsAsFactors=FALSE)

###########################################################################################################
#### Affectation de la poulation aux foyers

#Paramétres

#age referent +/- V.age
V.age <- 2
# classe d'age du partenaire par rapport a l'age du referent sélectionné
V.age.part <- 6
# classe d'age des enfants par rapport a l'age des parents
V.age.child <- 18
# probabilité d'avoir deux individus de meme genre ou de genre different au sein d'un couple sans enfant
# sur de la répartition renseignée dans les données de rencensement
Prob.gender.part <- c(12646+853,79+77)/sum(c(12646+853,79+77))
#IDEM pour les couples avec enfants
Prob.gender.part.ch <- c(11875+1560+930+10,5+0+11+0)/sum(c(11875+1560+930+10,5+0+11+0))
#probabilité relatives à la caractérstique 'travailleur' (O/N) pour les référent, partemaires et enfants
Prob.work.ref <-
(c(nrow(pop.ind.1[(pop.ind.1[,"worker"]=="No")&(pop.ind.1[,"companion"]=="Yes"),]),nrow(pop.ind.1[(pop.ind.1[
,"worker"]=="Yes")&(pop.ind.1[,"companion"]=="Yes"),])))/nrow(pop.ind.1[pop.ind.1[,"companion"]=="Yes",])
Prob.work.part<-(c(nrow(pop.ind.2[pop.ind.2[,"worker"]=="No",]),nrow(pop.ind.2[pop.ind.2[,"worker"]=="Yes",])
))/nrow(pop.ind.2)
Prob.work.child<-(c(nrow(pop.ind.3[pop.ind.3[,"worker"]=="No",]),nrow(pop.ind.3[pop.ind.3[,"worker"]=="Yes",]
)))/nrow(pop.ind.3)

pop.agents <- NULL

for (i in 1:nrow(pop.menages)){

  #Données necessaire pour condition 'travailleur'
  Nbworkers <- pop.menages[i,"Nbworkers"]
  if (pop.menages[i,"NbChilds"]>0){
  P <- pop.ind.3[pop.ind.3[,"age"] <= pop.menages[i,"Ref_Age"]-V.age.child-V.age.part,]
  P <- P[P[,"worker"] == "Yes",]
  can.child.work <- ifelse(nrow(P)>0,TRUE,FALSE)
  } else {
    can.child.work <- TRUE
  }

  ####REFERENT

  #Sélection d'un individu referent du ménage selon son age / companion et child
  P <- pop.ind.1[(pop.ind.1[,"age"] <= pop.menages[i,"Ref_Age"]+V.age) & (pop.ind.1[,"age"] >=
pop.menages[i,"Ref_Age"]-V.age),]
```

```
  is.companion <- ifelse(pop.menages[i,"NbAdults"]>1,"Yes","No")
  is.child <- ifelse(pop.menages[i,"NbChilds"]>0,"Yes","No")
  P <- P[(P[,"companion"]==is.companion)&(P[,"child"]==is.child),]

  #Condition "travailleur ?"
  if (pop.menages[i,"HHsize"] == Nbworkers){
    is.worker.ref <- "Yes"
  } else if (Nbworkers == 0){
    is.worker.ref <- "No"
  } else{
    if((is.companion == "No")&(!can.child.work)&(Nbworkers == 1)){
      is.worker.ref <- "Yes"
    }else if((!can.child.work)&(Nbworkers == 2)){
      is.worker.ref <- "Yes"
    } else{
    is.worker.ref <- ""
    }
  }
   if (is.worker.ref != ""){
    P <- P[(P[,"worker"]==is.worker.ref),]
  }

  #Sélection d'un candidat (> NORMALEMENT AU MOINS 1 CANDIDAT) si >2, si 1 et si 0
  if (nrow(P)>1){
  r <- sample(c(1:nrow(P)),1)
  pop.agents <- rbind(pop.agents,cbind(pop.menages[i,"id"],P[r,],pop.menages[,-1][i,]))
  Ref<- P[r,]
  } else if (nrow(P) == 1){
  pop.agents <- rbind(pop.agents,cbind(pop.menages[i,"id"],P[1,],pop.menages[,-1][i,]))
  Ref<- P[1,]
  } else{
    warning('pas de candidat pour le référent du ménage', pop.menages[i,"id"])
  }

  ####PARTENAIRE

  if (is.companion == "Yes"){

    # Genre du partenaire
    if (is.child=="Yes"){
      Prob <- Prob.gender.part.ch
    }else{
      Prob <- Prob.gender.part.ch
    }
    r<- sample(c(0,1),1,replace = FALSE, Prob)
    if (r==0){
      P <- pop.ind.2[!(pop.ind.2[,"gender"]==Ref[,"gender"]),]
    }else{
      P <- pop.ind.2[(pop.ind.2[,"gender"]==Ref[,"gender"]),]
    }

    # Age du partenaire et child
    P <- P[(P[,"age"] <= (as.numeric(Ref[,"age"]))+V.age.part) & (P[,"age"] >=
as.numeric(Ref[,"age"])-V.age.part),]
    is.child <- ifelse(pop.menages[i,"NbChilds"]>0,"Yes","No")
    P <- P[(P[,"child"]==is.child),]
    c(P[1,],10)
    # Travailleur ?
    is.worker.ref <- as.character(Ref[1,"worker"])
    Nbworkers <- ifelse(is.worker.ref == "Yes",Nbworkers-1,Nbworkers)

    if (pop.menages[i,"HHsize"]-1 == Nbworkers){
      is.worker.part <- "Yes"
    } else if (Nbworkers == 0){
      is.worker.part <- "No"
    } else{
      if((!can.child.work)&(Nbworkers == 1)){
        is.worker.part <- "Yes"
      } else {
```

```
      is.worker.part <- ""
      }
    }
    if (is.worker.part != ""){
      P <- P[(P[,"worker"]==is.worker.part),]
    }

  #Sélection d'un candidat (> NORMALEMENT AU MOINS 1 CANDIDAT) si >2, si 1 et si 0
    if (nrow(P)>1){
    r <- sample(c(1:nrow(P)),1)
    pop.agents <- rbind(pop.agents, cbind(pop.menages[i,"id"],P[r,],pop.menages[,-1][i,]))
    Part<- P[r,]
    } else if (nrow(P) == 1){
    pop.agents <- rbind(pop.agents, cbind(pop.menages[i,"id"],P[1,],pop.menages[,-1][i,]))
    Part<- P[1,]
    } else{
      warning('pas de candidat pour le partenaire pour ce ménage',pop.menages[i,"id"])
    }
  }

    ####ENFANT(S)

  if (is.child == "Yes") {

    is.worker.part <- as.character(Part[1,"worker"])
    Nbworkers <- ifelse(is.worker.part == "Yes",Nbworkers-1,Nbworkers)

    for (j in 1:pop.menages[i,"NbChilds"]){

      # Sélection des enfants selon l'age
      P <- pop.ind.3[pop.ind.3[,"age"] <= as.numeric(Ref[,"age"])-V.age.child,]

      # Travailleur
      if (Nbworkers == 0){
        is.worker.child <- "No"
      } else {
        is.worker.child <- "Yes"
        Nbworkers <- Nbworkers-1
      }
      P <- P[(P[,"worker"]==is.worker.child),]

      #Sélection d'un candidat (> NORMALEMENT AU MOINS 1 CANDIDAT) si >2, si 1 et si 0
      if (nrow(P)>1){
        r <- sample(c(1:nrow(P)),1)
        pop.agents <- rbind(pop.agents, cbind(pop.menages[i,"id"],P[r,],pop.menages[,-1][i,]))
      } else if (nrow(P) == 1){
        pop.agents <- rbind(pop.agents, cbind(pop.menages[i,"id"],P[1,],pop.menages[,-1][i,]))
      } else {
        warning('pas de candidat pour les enfants du ménage',pop.menages[i,"id"])
      }
    }
  }
}

#########################################################################################################
#Exportation de la table pop.agents
write.table(pop.agents,file=paste(Chemin.Data,"Save table/pop.agents.csv",sep=""),append=FALSE,
sep=";",row.names=FALSE,col.names=TRUE)
rm(list=ls())
#########################################################################################################
#Fin du module 3.1.
#########################################################################################################
```

MODULE 3.2. CREATION D'UNE POP. SYNTHETIQUE D'INDIVIDUS : Hill Climbing



```
#Module 3.2. CREATION D'UNE POPULATION SYNTHETIQUE D'INDIVIDU - méthode hill climbing adaptée
#
#Auteur : Pierre CUENCA
#27.07.2017
#-----------------------------------------------------------------------------------------------------------
# Applique une méthode de Hill climbing sur la population synthétique d'agents crée dans le module 3.1.
# afin que les caractéristiques de la population synthétique correnspondent aux caractéristiques (données
# de recensement) de la population réelle
#-----------------------------------------------------------------------------------------------------------
#Récupération des données des modules 2 et 3.2

#Specification du chemin du dossier
Chemin.Data <- "D:/_Travail/_TFE/_R.Bases de données/"

#Recuperation de la table pop.agents (pop synthetique d'agents créer dans le module 3.1.)
pop.agents<- read.table(paste(Chemin.Data,"Save
table/pop.agents.csv",sep=""),sep=";",header=TRUE,stringsAsFactors=FALSE)

#Recuperation des tables de 'candidats' crées dans le module 2.
pop.ind.1<- read.table(paste(Chemin.Data,"Save
table/pop.ind.1.csv",sep=""),sep=";",header=TRUE,stringsAsFactors=FALSE)
pop.ind.2<- read.table(paste(Chemin.Data,"Save
table/pop.ind.2.csv",sep=""),sep=";",header=TRUE,stringsAsFactors=FALSE)
pop.ind.3<- read.table(paste(Chemin.Data,"Save
table/pop.ind.3.csv",sep=""),sep=";",header=TRUE,stringsAsFactors=FALSE)

###########################################################################################################

#Recuperation et traitement des données de recensement pour les attributs d'individus

source('D:/_Travail/_TFE/_R.CodeTFE/traitement.donnee.census.R')
SecteurId <- c("62063A00-","62063A01-","62063A02-")

#age
Census.age <- read.table(paste(Chemin.Data,"Pour
export/Age.csv",sep=""),sep=";",header=TRUE,stringsAsFactors=FALSE)
Census.age <- traitement.donnee.census(Census.age[-length(Census.age)],SecteurId)

#genre
Census.gender <- read.table(paste(Chemin.Data,"Pour
export/Sexe.csv",sep=""),sep=";",header=TRUE,stringsAsFactors=FALSE)
Census.gender <- traitement.donnee.census(Census.gender[-length(Census.gender)],SecteurId)

#Degree

#Status

#equilibrage : ramener à 2909 ménages et 4070 individus
Census.age <- round(Census.age*nrow(pop.agents)/sum(Census.age),0)
Census.gender <- round(Census.gender*nrow(pop.agents)/sum(Census.gender),0)

###########################################################################################################
# 'Hill climbing' : vérifier les contraintes de recensement pour les agents
###########################################################################################################
#### Caracteristiques de la population synthétique

#Nombre de personnes par classes d'age pour la population sythetique
Agents.age <- NULL
for (p in 1:19){
  Agents.age <- c(Agents.age,nrow(pop.agents[(pop.agents[,"age"]<p*5)&(pop.agents[,"age"]>=(p-1)*5),]))
}
Agents.age <- c(Agents.age,nrow(pop.agents[pop.agents[,"age"]>=95,]))
#Sauvegarde des caractéristiques initiales
Agents.age.ini <- Agents.age

#Nombre de personnes selon le sexe dans la population synthtique
Agents.gender <-
c(nrow(pop.agents[pop.agents[,"gender"]=="Male",]),nrow(pop.agents[pop.agents[,"gender"]=="Female",]))
```

```
#Sauvegarde des caractéristiques initiales
Agents.gender.ini <- Agents.gender

#Status

#Degree

#########################################################################################################################
#### Cacul d'un score traduisant le caractère "présente de mauvaises caractéristiques" pour chaque agent
#Un score négatif pour la deuxieme valeur de score rends compte d'un caractère de 'bon' agent

score <- NULL

for (i in 1:nrow(pop.agents)){
  score.i.1 <- 0
  score.i.2 <- 0
#score pour le critere 'age'
  c.age <- min(c(floor(pop.agents[i,"age"]/5)+1,20))
    if (Agents.age[c.age] > Census.age[c.age]){
      score.i.1 <- score.i.1 + 1
  }
  score.i.2 <- score.i.2+Agents.age[c.age]-Census.age[c.age]

#score pour le critère 'gender'
  c.gender <- ifelse(pop.agents[i,"gender"]=="Male",1,2)
  if (pop.agents[i,"companion"]=="No"){   #On ne va pas changer le genre des individus dans les couples
    if (Agents.gender[c.gender] > Census.gender[c.gender]){
      score.i.1 <- score.i.1 +1
    }
    score.i.2 <- score.i.2+Agents.gender[c.gender]-Census.gender[c.gender]
  }
#Status

#Degree

   score <- rbind(score,c(score.i.1,score.i.2,i))
}

#########################################################################################################################
V.age.part <- 6 # classe d'age du partenaire par rapport a l'age du referent sélectionné (cf.module 3.1.)

#conditions d'arret
Diff.age <-Agents.age-Census.age
Cond.age <- sum(abs(Diff.age))
Diff.gender <- Agents.gender-Census.gender
Cond.gender <- sum(abs(Diff.gender))
e.age <- abs(sum(Agents.age)-sum(Census.age))
e.gender <- abs(sum(Agents.gender)-sum(Census.gender))
e.age.i0 <- e.age # pour analyse évaluation des performances
e.gender.i0 <- e.gender # pour analyse évaluation des performances

max.iter <- 3*max(Cond.age , Cond.gender)
iter<-0
converg<-0
Cond.age.pr <- Cond.age
Cond.gender.pr <- Cond.gender

#----DEBUTS DE LA BOUCLE-------------------------------------------------------------------------------------

while (((Cond.age > e.age) | (Cond.gender > e.gender)) & (converg < 50) & (iter <= max.iter) & (iter <=
2000)){

Cond.age.pr <- min(Cond.age,Cond.age.pr)
Cond.gender.pr <- min(Cond.gender,Cond.gender.pr)

####Remplacement du 'pire' agents/ou selection du 'meilleur' agents afin de compléter sa classe en priorité

#Sélection d'un candidat de plus mauvais score a remplacer / ou de meilleur score pour completer
  score.selec <- score[abs(score[,2])==max(abs(score[,2])),]
```

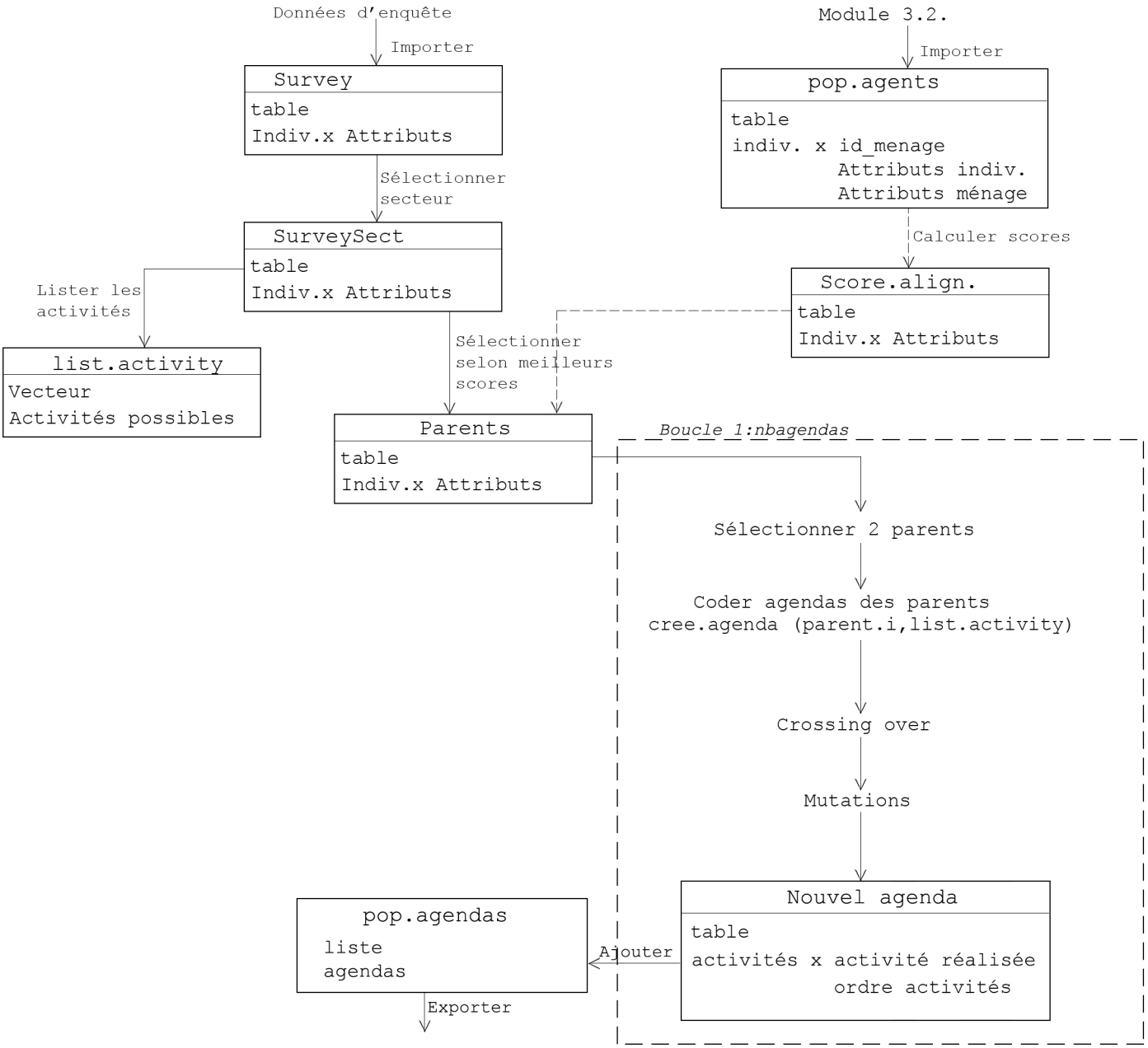
```
  if (nrow(score.selec)>1){
    r <- sample(c(1:nrow(score.selec)),1,replace=FALSE)
    ind <- score.selec[r,3]
  }else{
    ind <- score.selec[1,3]
  }

#-----------------------------------------------------------------------------------------------------------

#####Recherche d'un candidat que pourrait remplacer le meilleur agents
#si l'agent sélectionné est un 'bon' agents

if (score[ind,2]<0){

  #### recherches des individus que cet individu pourrait remplacer dans la table
  #l'individu doit avoir la meme position
  P<-pop.agents[pop.agents[,"position"]==pop.agents[ind,"position"],]
  #il doit avoir les meme attributs pour 'companion', 'child' et 'worker' (Y/N)

P<-P[(P[,"companion"]==pop.agents[ind,"companion"])&(P[,"child"]==pop.agents[ind,"child"])&(P[,"worker"]==pop
.agents[ind,"worker"]),]

  #### calcul des scores pour ces individus
    score.cand <- NULL

      for (i in 1:nrow(P)){
        score.i.1 <- 0
        score.i.2 <- 0
      #score pour le critere 'age'
        c.age <- min(c(floor(P[i,"age"]/5)+1,20))
        if (Agents.age[c.age] > Census.age[c.age]){
          score.i.1 <- score.i.1 + 1
        }
        score.i.2 <- score.i.2 + Agents.age[c.age]-Census.age[c.age]
      #score pour le critère 'gender'
        c.gender <- ifelse(P[i,"gender"]=="Male",1,2)
        if (pop.agents[i,"companion"]=="No"){
          if (Agents.gender[c.gender] > Census.gender[c.gender]){
            score.i.1 <- score.i.1 + 1
          }
          score.i.2 <- score.i.2 + Agents.gender[c.gender]-Census.gender[c.gender]
        }
        #Status

        #Degree
         score.cand <- rbind(score.cand,c(score.i.1,score.i.2,i))
      }
  #### sélection du pire individu
    if (! is.null(nrow(score.cand))){
      score.cand <- score.cand[score.cand[,1] == max(score.cand[,1]),]
    }
    if (! is.null(nrow(score.cand))){
      score.cand <- score.cand[score.cand[,2] == max(score.cand[,2]),]
    }

    if (is.null(nrow(score.cand))){
          ind.pire<-score.cand[3]
    } else {ind.pire <- score.cand[,3]}

    P <- P[ind.pire,]

    if (nrow(P)>1){
    remp <- sample(c(1:nrow(P)),1,replace=FALSE)
    } else {remp <- 1}

    P <- P[remp,]
    ind1 <- which(pop.agents[,1] == P[,1])
    ind2 <- which(pop.agents[ind1,2]==P[,2])
    ind3 <- which(pop.agents[ind1[ind2],3]==P[,3])
```

```
    ind <- ind1[ind2[ind3]][1]
}
#-----------------------------------------------------------------------------------------------------------

#### Création de la table de candidats
  #sélection meme 'position'
  if (pop.agents[ind,"position"]=="Reference person HH") {
    P<-pop.ind.1
  } else if (pop.agents[ind,"position"]=="Partner"){
    P<-pop.ind.2
  } else {
    P<-pop.ind.3
  }

  #Sélection memes status pour 'companion', 'child' et 'worker' (Y/N)
  P <-
P[(P[,"companion"]==pop.agents[ind,"companion"])&(P[,"child"]==pop.agents[ind,"child"])&(P[,"worker"]==pop.ag
ents[ind,"worker"]),]
  #Controle sur l'âge des personnes en couple et avec enfants
  P <-
P[(P[,"companion"]==pop.agents[ind,"companion"])&(P[,"child"]==pop.agents[ind,"child"])&(P[,"worker"]==pop.ag
ents[ind,"worker"]),]
  #Controle sur l'âge des personnes en couple et avec enfants
  if ((pop.agents[ind,"companion"]=="Yes")&(pop.agents[ind,"child"]=="No")){
    if (pop.agents[ind,"position"]=="Reference person HH"){
      P<-P[(P[,"age"]<=pop.agents[ind+1,"age"]+V.age.part)&(P[,"age"]>=pop.agents[ind+1,"age"]-V.age.part),]
    } else {
      P<-P[(P[,"age"]<=pop.agents[ind-1,"age"]+V.age.part)&(P[,"age"]>=pop.agents[ind-1,"age"]-V.age.part),]
    }
  }
  if (pop.agents[ind,"child"]=="Yes"){
    if(pop.agents[ind,"companion"]=="Yes"){
      if(pop.agents[ind,"position"]=="Reference person HH"){

P<-P[(P[,"age"]<=pop.agents[ind+1,"age"]+V.age.part)&(P[,"age"]>=max(pop.agents[(ind+2):(ind+1+pop.agents[ind
,"NbChilds"]),"age"])+18),]
      }else{

P<-P[(P[,"age"]<=pop.agents[ind-1,"age"]+V.age.part)&(P[,"age"]>=max(pop.agents[(ind+1):(ind+pop.agents[ind,"
NbChilds"]),"age"])+18),]
      }
    } else {
      P<-P[P[,"age"]>=max(pop.agents[(ind+1):(ind+pop.agents[ind,"NbChilds"]),"age"])+18,]
    }
  }

  if (pop.agents[ind,"position"]=="Child"){
    P<-P[P[,"age"]<= pop.agents[ind,"Ref_Age"]-18,]
  }
  #On ne change pas le genre dans les couples
    if (pop.agents[ind,"companion"]=="Yes"){
      P<-P[P[,"gender"]==pop.agents[ind,"gender"],]
    }

  #sélection des agents avec le meilleur score
    #Calcul scores des cadidats
      score.cand <- NULL

      for (i in 1:nrow(P)){
        score.i.1 <- 0
        score.1.2 <- 0
      #score pour le critere 'age'
        c.age <- min(c((floor(P[i,"age"]/5)+1),20))
        if (Agents.age[c.age] > Census.age[c.age]){
          score.i.1 <- score.i.1 + 1
        }
        score.i.2 <- score.i.1 + Agents.age[c.age] - Census.age[c.age]

      #score pour le critère 'gender'
```

```
        c.gender <- ifelse(P[i,"gender"]=="Male",1,2)
        if (Agents.gender[c.gender] > Census.gender[c.gender]){
          score.i.1 <- score.i.1 +1
        }
        score.i.2 <- score.i.2 + Agents.gender[c.gender] - Census.gender[c.gender]

      #Status

      #Degree

         score.cand <- rbind(score.cand,c(score.i.1,score.i.2,i))
      }

####Sélection des meilleurs candidats
      if (! is.null(nrow(score.cand))){
        score.cand <- score.cand[score.cand[,1] == min(score.cand[,1]),]
      }
      if (! is.null(nrow(score.cand))){
        score.cand <- score.cand[score.cand[,2] == min(score.cand[,2]),]
      }

      if (is.null(nrow(score.cand))){
        ind.meil<-score.cand[3]
      } else {ind.meil <- score.cand[,3]}

      P <- P[ind.meil,]

      if (nrow(P)>1){
        remp <- sample(c(1:nrow(P)),1,replace=FALSE)
      } else {remp <- 1}

####Preparation du remplacement
#Mise a jour des tables de caractéristiques de la population synthetique
  #Age
    c.age <- min(c(floor(pop.agents[ind,"age"]/5)+1,20))
    Agents.age[c.age]<-Agents.age[c.age]-1
    c.age <- min(c(floor(P[remp,"age"]/5)+1,20))
    Agents.age[c.age]<-Agents.age[c.age]+1
  #Genre
    c.gender <- ifelse(pop.agents[ind,"gender"]=="Male",1,2)
    Agents.gender[c.gender] <- Agents.gender[c.gender]-1
    c.gender <- ifelse(P[remp,"gender"]=="Male",1,2)
    Agents.gender[c.gender] <- Agents.gender[c.gender]+1
  #Degree

  #Status

#### Remplacement de l'agent
    pop.agents[ind,] <- cbind(pop.agents[ind,1],P[remp,],pop.agents[,(-ncol(P)-1):-1][ind,])

#### Calcul des nouveaux scores

score <- NULL

for (i in 1:nrow(pop.agents)){
  score.i.1 <- 0
  score.i.2 <- 0
#score pour le critere 'age'
  c.age <- min(c(floor(pop.agents[i,"age"]/5)+1,20))
  if (Agents.age[c.age] > Census.age[c.age]){
      score.i.1 <- score.i.1 + 1
  }
  score.i.2 <- score.i.2+Agents.age[c.age]-Census.age[c.age]

#score pour le critère 'gender'
  c.gender <- ifelse(pop.agents[i,"gender"]=="Male",1,2)
  if (pop.agents[i,"companion"]=="No"){   #On ne va pas changer le genre des individus dans les couples
    if (Agents.gender[c.gender] > Census.gender[c.gender]){
      score.i.1 <- score.i.1 +1
```

```
    }
    score.i.2 <- score.i.2+Agents.gender[c.gender]-Census.gender[c.gender]
  }
#Status

#Degree

   score <- rbind(score,c(score.i.1,score.i.2,i))
}

Diff.age <-Agents.age-Census.age
Cond.age <- sum(abs(Diff.age))
Diff.gender <- Agents.gender-Census.gender
Cond.gender <- sum(abs(Diff.gender))

if((Cond.age >= Cond.age.pr) & (Cond.gender >= Cond.gender.pr) ){
  converg <- converg + 1
} else {converg <-0}

iter<-iter+1

#print(iter) #permet de suivre l'avancé du processus
}
#----FIN DE LA BOUCLE-----------------------------------------------------------------------------------------------




#########################################################################################################################
#COMPARAISON CARACTERISTIQUES POP SYNTH.

Census.age
Agents.age.ini
Agents.age
Agents.age-Census.age

e.age.i0
e.age

Census.gender
Agents.gender.ini
Agents.gender

e.gender.i0
e.gender

#########################################################################################################################
#Enregistrement de la table pop.agents
write.table(pop.agents,file=paste(Chemin.Data,"Save table/pop.agents.m22.csv",sep=""),append=FALSE,
sep=";",row.names=FALSE,col.names=TRUE)
rm(list=ls())
#########################################################################################################################
# Fin module 3.2.
#########################################################################################################################
```

MODULE 4. CREATION D'AGENDA : création d'une population initiale pour un algorythme génétique



```
#Module 4. CREATION D'AGENDA : création d'une population initiale pour un algorythme génétique
#
#Auteur : Pierre CUENCA
#30.07.2017
#-----------------------------------------------------------------------------------------------------------
# Pour un individu de la population synthetique le but est de trouvé des individus enquétés avec un profil
# proche et de croiser et muter leurs agendas observées afin de créer une population servant de base
# pour un alogorithme génétique tel que celui proposé par Charypar et Nagel
#
# Remarque 1 : les agendas des données d'enquête dont je dispose ne contiennent pas de données horaires
# Remarque 2 : nous ne prendrons pas en compte ici non plus la localisation des activités
# Toutefois, ces deux dimensions essentielles des agendas et également le mode de transport pourraient être
# intégrer dans la démarche méthodologique proposée ici
#-----------------------------------------------------------------------------------------------------------
# Récupération de la population synthetique et des données d'enquête

#Specification du chemin du dossier
Chemin.Data <- "D:/_Travail/_TFE/_R.Bases de données/"

source('D:/_Travail/_TFE/_R.CodeTFE/cree.agenda.R')

#Population synthetique crée dans le module 3.2.
pop.agents<- read.table(paste(Chemin.Data,"Save
table/pop.agents.m22.csv",sep=""),sep=";",header=TRUE,stringsAsFactors=FALSE)

#Données d'enquête
load(paste(Chemin.Data,"pierre.rda",sep=""))
Survey <- pierre.dat[complete.cases(pierre.dat),]

###########################################################################################################
#### Sélection d'un secteur géographique

Secteur <- 62063    # Ville de Liège (code des secteurs statistiques)

SurveySect <- NULL

for (i in 1:nrow(Survey)){
  if (is.na(Survey[i,27]) == FALSE){     # s'il ya des valeurs NA dans le tableau
    if (Survey[i,27] == Secteur)
    {SurveySect <- rbind(SurveySect,Survey[i,])
    }
  }
}
rm(Survey,pierre.dat,i,Secteur)
###########################################################################################################
#### Codage des agendas a partir des agendas

    # 1: P: déposer/chercher quelqu'un
    # 2: H: aller à la maison
    # 3: W: aller travailler
    # 4: B: pour le travail (si tournée, nombre : déplacements)
    # 5: E: suivre un cours (école, .)
    # 6: R: prendre un repas
    # 7: S: faire des courses/du shopping
    # 8: F: services (médecin, banque, .)
    # 9: V: rendre visite à la famille ou à des amis
    # 10: T: se promener, faire un tour
    # 11: L: loisirs, sports, culture
    # 12: O: autre (précisez)

# création de la table des activités possibles
# chaque activité doit etre 'unique' il faut ainsi differncier par exemple travailler le matin et travail
l'apres-midi
# voire plus de classe encore pour certains inividus
# On va donc se baser sur l'ensemble des observations faites.

comp<-rep(0,12)
activities <- c("P","H","W","B","E","R","S","F","V","T","L","O")
```

```
for (i in 1:nrow(SurveySect)){
  seq.i <- unlist(strsplit(SurveySect[i,"seq"],split=""))

  #Suppression de répétitions 'inexpliquées'
  #pour H,W,R et E
  a.norep <- c("H","W","R","E")
  ind.supp<-NULL
  if (length(seq.i)>1){
    for (k in 1:(length(seq.i)-1)){
        if ((seq.i[k]==seq.i[k+1])&(sum(a.norep%in%seq.i[k])>0)){
          ind.supp <- c(ind.supp,k)
        }
    }
    if (! is.null(ind.supp)){
      seq.i<-seq.i[-ind.supp]
    }
  }
  #nombre max d'occurence des activités
  for (p in 1:12){
  s <- sum(seq.i%in%activities[p])

## 2 autres solutions
#   G<-grep(activities[p], seq.i)
#   length(G)
#
#   seq.i <- SurveySect[i,"seq"]
#   length(gregexpr(activities[p],seq.i)[[1]])
##
  comp[p]<-max(comp[p],s)
  }
}

list.activity <- NULL
for (i in 1:length(comp)){
  if (comp[i]>0){
    for (j in 1:comp[i]){
      list.activity <- c(list.activity,activities[i])
    }
  }
}
rm(i,j,k,p,s,seq.i,comp,ind.supp)
#############################################################################################################
#CREATION D'UNE POPULATION INITIALE D'AGENDA POUR UN AGENT DE LA POPULATION SYNTHETIQUE

####Dans notre cas on va travailler avec le profil d'un agent sélectionné aléatoirement
    ID.Agent <- sample(c(1:nrow(pop.agents)),1,replace=FALSE)
    agent <- pop.agents[ID.Agent,]

####Calcul d'un score de ressemblance afin de selectionner un groupe de "parents" parmi les individus
enquetés

  # Je fais ici une proposition trés simple, il est possible d'établir une fonction plus élaborée
  # sans remettre en cause tout le principe pour autant
    ## Critère
    #identité > 1
    #difference > 0
    ## Ponderation
    #age(3),gender(2),position(2),degree(1),status(1),companion(2),child(2),worker(2),HHsize(1),HHincome(1)
    #score max = 17

  score.table <- NULL

  for (i in 1:nrow(SurveySect)){

    score <- 0
  # pour les critere les plus simples (test d'égalié)
    colnames.crit<- c("gender","position","degree","status","companion","child")
    incr.score <- c(2,2,1,1,2,2)
```

```
    for (k in 1:6){
      if(SurveySect[i,colnames.crit[k]]==agent[1,colnames.crit[k]]){
        score <- score + incr.score[k]
      }
    }
  #HHsize/hhsize
    if(SurveySect[i,"hhsize"]==agent[1,"HHsize"]){
        score <- score + 1
    }
  #HHincome/hhicome
    if(SurveySect[i,"hhincome"]==agent[1,"HHincome"]){
        score <- score + 1
    }
  #age > memes classes d'age (5ans ->1pts et 0-18,18-30,30-45,45-60,60-75,>75 ->2pts)
    #classes d'age - 5ans
      c.age.agent <- min(c(floor(agent[1,"age"]/5),20))
      c.age.ind.i <- min(c(floor(SurveySect[i,"age"]/5),20))
    if(c.age.agent==c.age.ind.i){
      score <- score +1
    }
    #classes étendues - 15 ans
    if(agent[1,"age"] < 18){ c.age.agent <- 0 }
    c.age.agent <- min(c(floor(agent[1,"age"]/15),6))
    if(SurveySect[i,"age"] < 18){ c.age.agent <- 0 }
    c.age.ind.i <- min(c(floor(SurveySect[i,"age"]/15),6))

    if(c.age.agent==c.age.ind.i){
      score <- score +2
    }
  #worker (Y/N)
    worker<-"Yes"
    if((SurveySect[i,"status"]=="Pupil, student") | (SurveySect[i,"status"]=="(Pre)retired")|
(SurveySect[i,"status"]=="Housewife/househusband")| (SurveySect[i,"status"]=="Unemployed")|
(SurveySect[i,"status"]=="Incapacitated")){
      worker <- "No"
    }

    if (agent[1,"worker"] == worker){
      score <- score + 2
    }
    score.table <- rbind(score.table,c(i,score))
  }

    rm(i,k,incr.score,score,worker,c.age.agent,c.age.ind.i,colnames.crit)

#### Sélections des individus "parents"
      Nbparents <- 6 #parametre

      score.selec <- max(score.table[,2])
      Cand.parent <- NULL
      nbcand<-0

    while (nbcand <Nbparents){
      # Recuperation des candidats sélectionné dans la table Cand.parent
      ind.selec <- score.table[score.table[,2]==score.selec,]
      if(is.null(nrow(ind.selec))){
        ind.selec <- ind.selec[1]
      }else{
        ind.selec <- ind.selec[,1]
      }
      Cand.parent <- rbind(Cand.parent,SurveySect[ind.selec,])

      # Retire les candidats sélectionnés à l'étape precedente
      score.table <- score.table[!(score.table[,2]==score.selec),]

      # Maj des parametres de l'itéraction
      score.selec <- max(score.table[,2])
      nbcand<-nrow(Cand.parent)
    }
```

```
    if (nbcand >6){
      R <- sample(1:nbcand,6,replace=FALSE)
      Parents <- Cand.parent[R,]
    } else {
      Parents <- Cand.parent
    }
    rm(Cand.parent,score.selec,nbcand,ind.selec)


#########################################################################################################################
#### Création de la poplulation d'agendas
    n.agendas.T <- 20  #parametre

    list.agendas <- list()
    nbagenda <-0

    ##Création des agendas pour les parents a partir de la sequence activités (donnée d'enquête)
    list.agenda.p <- list()
      for (k in 1:Nbparents){
        list.agenda.p  <- c(list.agenda.p, list(cree.agenda(Parents[k,"seq"],list.activity)))
      }

    list.agendas <- list()
    n.agendas<-0

  while (n.agendas < n.agendas.T){

      ## Sélection de deux parents
        R <- sample(1:nrow(Parents),2,replace=FALSE)
        parents.i <- Parents[R,]

      ## recuperation des agendas pour simplicité d'écriture et de lecture du code
        agenda.1 <- list.agenda.p[[R[1]]]
        agenda.2 <- list.agenda.p[[R[2]]]

      ##Initialisation : création d'un agenda 'vide'
        is.activity <- rep(0,length(list.activity))
        order.activity <- rep(0,length(list.activity))

        agenda <- cbind(list.activity,as.data.frame(is.activity),as.data.frame(order.activity))

      ## Crossing-over

        for (n in 1:nrow(agenda)){
          if (agenda.1[n,2]==agenda.2[n,2]){
            agenda[n,2]<- agenda.1[n,2]
            agenda[n,3]<- agenda[n,2]*(agenda.1[n,3]+agenda.2[n,3])/2
          } else{
            r <- sample(0:1,1,replace=FALSE)
            agenda[n,2]<- r
            if (r==1){
              agenda[n,3]<- agenda.1[n,3]+agenda.2[n,3]
            } else{
              agenda[n,3]<-0
            }
          }
        }

        if (sum(agenda[,3])>0){
          O <- order((agenda[which(agenda[,3]!=0),3])*2)
          for (i in 1:length(O)){
            agenda[O[i],3]<-i
          }
        }

      ## Mutations
        prob<-c(1-1/(2*length(list.activity)), 1/(2*length(list.activity))) #peut etre définie selon le type
d'activité
```

```
        for (n in 1:nrow(agenda)){
          mut<-sample(c(FALSE,TRUE),1,replace=FALSE,prob)
          if (mut){
            if(agenda[n,2]==0){
              agenda[n,2]<-1

              ord <- sample(1:(sum(!agenda[,3]%in%0)+1),1,replace=FALSE)
              agenda[n,3]<- ord

              W <- which(agenda[,3]!=0)
              W <- W[-which(W==n)]
              W2 <- which(agenda[W,3]>=ord)
              ind <- W[W2]
              agenda[ind,3]<-agenda[ind,3]+1
            } else{
              agenda[n,2]<-0
              ord <- agenda[n,3]
              agenda[n,3]<-0

              W <- which(agenda[,3]!=0)
              W2 <- which(agenda[W,3]>=ord)
              ind <- W[W2]
              agenda[ind,3]<- agenda[ind,3]-1
            }
          }
        }

      ## Pour se ramener a un codage comme celui de Charypar et Nagel
        # l'ordre des activité doit etre déterminer pour l'ensemble des activité potentillement réalisables
        # le but est de donner définir un 'moment de rélisation'(ordre) pour les activités non réalisées
        # tout en conservant l'ordre des activité réalisée

          for (k in 1:(length(which(agenda[,3]==0)))){
            W <- which(agenda[,3]==0)
            n <- W[sample(1:length(W),1,replace=FALSE)]
            ord <- sample(1:(sum(!agenda[,3]%in%0)+1),1,replace=FALSE)
            agenda[n,3]<- ord

            W <- which(agenda[,3]!=0)
            W <- W[-which(W==n)]
            W2 <- which(agenda[W,3]>=ord)
            ind <- W[W2]
            agenda[ind,3]<- agenda[ind,3]+1
          }

      list.agendas <- c(list.agendas,list(agenda))
      n.agendas <- length (list.agendas)
    }

#########################################################################################################
#Exportation de la liste d'agendas
  ## Mise en forme
    pop.agendas<-list.agendas[[1]]
  for(i in 2:length (list.agendas)){
    pop.agendas<- cbind(pop.agendas,list.agendas[[i]])
  }
  ## Exportattion
file.name <- paste(paste(Chemin.Data,"Save
table/pop.agendas.",sep=""),paste(as.character(ID.Agent),".csv",sep=""),sep="")

write.table(pop.agendas,file=file.name,append=FALSE, sep=";",row.names=FALSE,col.names=TRUE)
rm(list=ls())
#########################################################################################################
# Fin du module 4.
#########################################################################################################
```

```
#Fonction traitement.donnee.census
#
#Auteur: Pierre CUENCA
#-------------------------------------------------------------------------------
# Permet d'extraire des tables les valeurs des secteurs sélectionnés et de les agréger
#-------------------------------------------------------------------------------

traitement.donnee.census <- function(data,secteur)

# data : table de données (dim2 : Secteurs x données)
# secteur : vecteur de chaines de caractères (code des secteurs ex: "62063A0-")

{
  SecteurId <- secteur
  HHsize0 <- data

#Initialisation
  k <- 1
while (HHsize0[k,1] != SecteurId[1] | k == nrow(HHsize0)) {
  k<-k+1
}

if (k<= nrow(HHsize0)){
  HHsizesect <- HHsize0[k,]
  } else {
print("erreur secteur manquant")
}

#Iteration copie les données pour chaque secteur de la zone considérée
for (i in 2:length(SecteurId)){
k <- 1
  while (HHsize0[k,1] != SecteurId[i]| k == nrow(HHsize0)) {
      k <- k+1
  }
if (k<= nrow(HHsize0)){
   HHsizesect <- rbind(HHsizesect,HHsize0[k,])}
else {print("erreur secteur manquant")
}
}

#Somme des valeurs des secteurs sélectionnés

  HHsizesect <- HHsizesect[,-1]

HHsizenum <- matrix(nrow=nrow(HHsizesect), ncol=ncol(HHsizesect))

for (j in 1:ncol(HHsizesect)){
  for (i in 1:nrow(HHsizesect)){
    if (is.character(HHsizesect[i,j])){
      u <- as.numeric(sub(" ","",HHsizesect[i,j]))
    }
      else {u <- as.numeric(HHsizesect[i,j])}
    HHsizenum[i,j] <- u
  }
}

k <- 1
HHsize <- sum(HHsizenum[,k])
k <- k+1

  #sum(HHsizesect[,k]

for (i in k:ncol(HHsizesect)){
  HHsize <- c(HHsize, sum(HHsizenum[,i]))
}

return (HHsize) #renvoie HHsize
}
```

```
# IPF multidimentionel
#
#Auteur: Pierre CUENCA
#d'après la fonction Ipfp de Johan Barthelemy and Thomas Suesse
#20.07.2017
#------------------------------------------------------------------------------
#Ce fichier fourni la fonction ifpf qui implémente
#une méthode d'ajustement proportionnel par iterations
#------------------------------------------------------------------------------
ipfp <- function(seed,target.list,target.data,iter, tol){

#seed correspond au tableau multidimentionel à ajuster
#target.list
#target.data est une liste de vecteur de cibles marginales
#iter le nombre d'iteration
#tol la valeur de tolerence associée à la condition d'arrêt


# initial value is the seed
  result <- seed
  converged <- FALSE

for (n in 1:iter){

  # sauve précédente itération pour tester la convergence
  result.prev <- result

  ##Boucle sur les contraintes
  for (i in 1:length(target.data)){
    # extraction des marges actuelles
    sum.temp <- apply(result,target.list[[i]],sum)
    # calcul du facteur à appliquer
    update.factor <- target.data[[i]]/sum.temp
    # applique le facteur
    result <- sweep(result, target.list[[i]], update.factor, FUN= "*")
  }

  ## Verification de la condition d'arrêt
    stp.crit <- max(abs(result - result.prev))
    if (stp.crit < tol) {
      converged <- TRUE
    cat('convergence apres', n,'iterations')
      break
    }
}

  # Verifie la convergence
  if (converged == FALSE) {
    warning('l IPFP n a pas convergé apres', iter, ' iteration(s)!')
  }

  # computing the proportions
  # result.prop <- result / sum(result)

  # gathering the results in a list
  # results.list <- list("x.hat" = result, "p.hat" = result.prop)

  # returning the result
  return(result)
}
```

```
#création de la diversité
#
#Auteur: Pierre CUENCA
#25.07.2017
#------------------------------------------------------------------------------------------------------------
#permet de créer plusieurs génération d'individus
```

```
#la fonction renvoie les individus de l'ensemble des générations créees ainsi que les individus de départ
#-----------------------------------------------------------------------------------------------------------
pop.gene <- function (M,n.gene){

# M : table des individus de depart
# n.gene : nombre de generations

  pop <- M
  pop.n <- M

  id <- 1

   for (n in 1:n.gene){
     pre <- paste("g", as.character(id),sep="")
     pop.n <- generation(pop.n, 2*nrow(pop.n), pre)
     pop <- rbind(pop, pop.n )
     id<-id+1
   }

return(pop)
}
```

```
#création de la diversité - variante étudiants
#
#Auteur: Pierre CUENCA
#25.07.2017
#-----------------------------------------------------------------------------------------------------------
#permet de créer plusieurs génération d'individus
#la fonction renvoie les individus de l'ensemble des générations créees ainsi que les individus de départ
#-----------------------------------------------------------------------------------------------------------
pop.gene.student <- function (M, n.gene){

# M : table des individus de depart
# n.gene : nombre de generations

  pop <- M
  pop.n <- M

  id <- 1

   for (n in 1:n.gene){
     pre <- paste("g", as.character(id),sep="")
     pop.n <- generation.student(pop.n, 2*nrow(pop.n), pre)
     pop <- rbind(pop, pop.n )
     id<-id+1
   }

return(pop)
}
```

```
#Fonction génération
#
#Auteur: Pierre CUENCA
#25.07.2017
#-----------------------------------------------------------------------------------------------------------
#Crée une génération fille à partir d'un groupe d'individu
#-----------------------------------------------------------------------------------------------------------
generation <- function (M, n, pre) {

# M : table des individus de départ
# n : nombre d'individus à générer
# pre : prefixe d'identification des individus de la génération produite

pop<-NULL
```

```
id<-1

for (p in 1:n){

  if(nrow(M)>1){

# choix de deux individus au hasard
R <- sample (c(1:nrow(M)),2,replace=FALSE)

#### Crossing over

new.indiv <- paste(paste(pre,"_",sep=""),as.character(id),sep="")

#Age
  r<-sample(R,1,replace=FALSE)
  new.indiv <- c(new.indiv, M[r,"age"] )
#Gender
  r<-sample(R,1,replace=FALSE)
  new.indiv <- c(new.indiv, M[r,"gender"] )
#position
  r<-sample(R,1,replace=FALSE)
  new.indiv <- c(new.indiv, M[r,"position"] )
#Degree-Status
  r<-sample(R,1,replace=FALSE)
  new.indiv <- c(new.indiv, M[r,"degree"],M[r,"status"] )
#Companion
  r<-sample(R,1,replace=FALSE)
  new.indiv <- c(new.indiv, M[r,"companion"] )
#Child
  r<-sample(R,1,replace=FALSE)
  new.indiv <- c(new.indiv, M[r,"child"] )
}
  else if (nrow(M) == 1){
    new.indiv<-c(paste(paste(pre,"_",sep=""),as.character(id),sep=""), M)
  }
  else{
  warning("il n'y a pas d'individu dans cette classe")
}
id<-id+1

#### Mutation
# on effectue des mutations sur l'age et le genre

#age
  mut <- sample (c(0:1),1,replace=FALSE,c(0.9,0.1))
if (mut == 1){
    new.indiv[2] <- sample(c(min(M[,"age"]):max(M[,"age"])),1)
}
#genre
mut <- sample (c(0:1),1,replace=FALSE,c(0.9,0.1))
if (mut == 1){
    new.indiv[3] <- sample(c("Male","Female"),1)
}

pop <- rbind(pop,new.indiv)
  }
colnames(pop)<-c("i_id","age","gender","position","degree","status","companion","child")
return (pop)
}
```

```
#Fonction génération.student
#
#Auteur: Pierre CUENCA
#25.07.2017
#-----------------------------------------------------------------------------------------------------
# Crée une génération fille à partir d'un groupe d'individu
#
# pour les classes d'étudiants, on observe une correlation age et degree/status
```

```
# (on ne peut pas avoir d'élève en primaire de 30 ans !!)
# il faudrait séparer en différentes classes pupil et student
# éventuellement il est possible de grouper actifs et étudiants !!
#-----------------------------------------------------------------------------------------------------------
generation.student <- function (M, n, pre){

# M : table des individus de départ
# n : nombre d'individus à générer
# pre : prefixe d'identification des individus de la génération produite

pop<-NULL
id<-1

for (p in 1:n){

  if(nrow(M)>1){

# choix de deux individus au hasard
R <- sample (c(1:nrow(M)),2,replace=FALSE)

# Crossing over

new.indiv <- paste(paste(pre,"_",sep=""),as.character(id),sep="")

#Age - degree - status
  r<-sample(R,1,replace=FALSE)
  r.save <- r
  new.indiv <- c(new.indiv, M[r,"age"] )
#Gender
  r<-sample(R,1,replace=FALSE)
  new.indiv <- c(new.indiv, M[r,"gender"] )
#position
  r<-sample(R,1,replace=FALSE)
  new.indiv <- c(new.indiv, M[r,"position"] )
#Degree-Status
  new.indiv <- c(new.indiv, M[r.save,5], M[r.save,"status"] )
#Companion
  r<-sample(R,1,replace=FALSE)
  new.indiv <- c(new.indiv, M[r,"companion"] )
#Child
  r<-sample(R,1,replace=FALSE)
  new.indiv <- c(new.indiv, M[r,"child"] )
} else if(nrow(M)==1){
    new.indiv <- c(paste(paste(pre,"_",sep=""),as.character(id),sep=""),M[,-1])
} else {
  warning("il n'y a pas d'individu dans cette classe")
}
id<-id+1

#### Mutation
# on effectue des mutations sur l'age et le genre

#age

mut <- sample (c(0,1),1,replace = FALSE, c(0.9,0.1))
if (mut == 1){

#recherches ages pour le meme "degree"

  Val.mut <- NULL

  for (i in 1:nrow(M)){
    if (M[i,5] == new.indiv[5]){
    Val.mut <- c(Val.mut, as.numeric(M[i,"age"]))
    }
  }

  min.age <- min(Val.mut)
  max.age <- max(Val.mut)
```

```
#modification de l'age dans une tranche observée
  if (min.age < max.age){
  new.indiv[2] <- sample(c(min.age:max.age), 1, replace = FALSE)

  }
}

#genre
  mut <- sample (c(0:1),1,replace=FALSE,c(0.7,0.3))
if (mut == 1){
    new.indiv[3] <- sample(c("Male","Female"),1)
}

 pop <- rbind(pop, new.indiv)
}

colnames(pop)<-c("i_id","age","gender","position","degree","status","companion","child")
return (pop)
}
```

```
#Fonction pour le codage des agendas
#
#Auteur : Pierre CUENCA
#30.07.2017
#-----------------------------------------------------------------------------------------------------------
# permet de transformer la séquence des données d'enquéte sous une forme exploitable
# On s'inspire de la méthode de codage des agendas de Charypar et Nagel
# un agenda sera dans notre cas deux tables : la premiere dertermine les activités menées
# la seconde détermine leur ordre.
#-----------------------------------------------------------------------------------------------------------

cree.agenda <- function(Seq, list.activity){

      # 1: P: déposer/chercher quelqu'un
      # 2: H: aller à la maison
      # 3: W: aller travailler
      # 4: B: pour le travail (si tournée, nombre :      déplacements)
      # 5: E: suivre un cours (école, .)
      # 6: R: prendre un repas
      # 7: S: faire des courses/du shopping
      # 8: F: services (médecin, banque, .)
      # 9: V: rendre visite à la famille ou à des amis
      # 10: T: se promener, faire un tour
      # 11: L: loisirs, sports, culture
      # 12: O: autre (précisez)

  seq<-unlist(strsplit(Seq,split="")) #vecteur de caractere

  ##Suppression de répétitions 'inexpliquées'
  #pour H,W,R et E
  a.norep <- c("H","W","R","E")
  ind.supp<-NULL
  if (length(seq)>1){
    for (k in 1:(length(seq)-1)){
        if ((seq[k]==seq[k+1])&(sum(a.norep%in%seq[k])>0)){
          ind.supp <- c(ind.supp,k)
        }
    }
    if (! is.null(ind.supp)){
      seq<-seq[-ind.supp]
    }
  }

  ## is.activity est un Vecteur binaire -> activité menée ou non
  ## order.activity est un Vecteur de permutation -> ordre des activités

  is.activity <- rep(0,length(list.activity))
```

```
  order.activity <- rep(0,length(list.activity))

  agenda <- cbind(list.activity,as.data.frame(is.activity),as.data.frame(order.activity))

  for (i in 1:length(seq)){

    act <- seq[i]

    W <-which (agenda[,1]==act)
    cond <- TRUE
    n <- 1
    while(cond){
      if(agenda[W[n],2]==0){
        # activité reéalisées
        agenda[W[n],2]<-1
        # ordre des activités
        agenda[W[n],3]<-i
        #condition de fin de la boucle while
        cond <- FALSE
      }
      n<-n+1
    }
  }
############################################################
  # l'ordre des activité doit etre déterminer pour l'ensemble des activité potentillement réalisables
  # le but est de donner définir un 'moment de rélisation'(ordre) pour les activités non réalisées
  # tout en conservant l'ordre des activité réalisée
        #
        #   for (k in 1:(length(list.activity)-length(seq))){
        #     W <- which(agenda[,3]==0)
        #     n <- W[sample(1:length(W),1,replace=FALSE)]
        #     ord <- sample(1:(sum(!agenda[,3]%in%0)+1),1,replace=FALSE)
        #     agenda[n,3]<- ord
        #
        #     W <- which(agenda[,3]!=0)
        #     W <- W[-which(W==n)]
        #     W2 <- which(agenda[W,3]>=ord)
        #     ind <- W[W2]
        #     agenda[ind,3]<- agenda[ind,3]+1
        #   }
############################################################

 return(agenda)
}
```