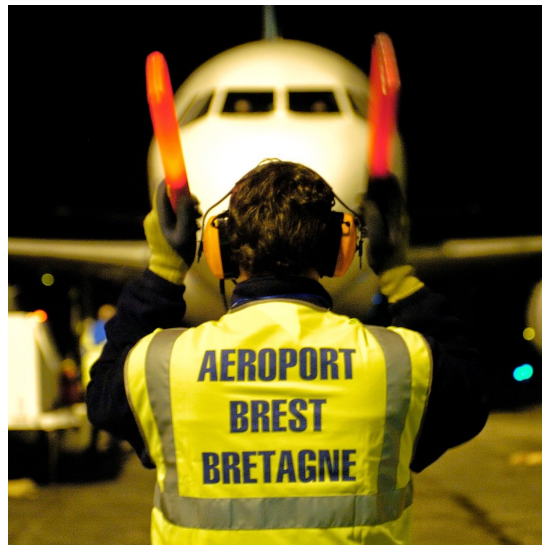


RAPPORT DE PROJET SIMULATION

UV 5.8 Simulation

Simulation de l'aéroport de Brest



Rédigé par :

Alice Danckaers
Thomas Boulier

Sous la direction de :

Olivier Verron



ENSTA
Bretagne

Option Systèmes Perception Information Décision

© 2016 Alice Danckaers and Thomas Boulier.

Licensed under the Creative Commons Attribution-ShareAlike 4.0 International Public License.

Sommaire

1	Objectifs de l'étude	4
2	Analyse du problème	5
2.1	L'aéroport	6
2.2	Les avions	6
2.3	Les infrastructures	6
2.4	Le scénario	6
2.5	Les listes d'attente	6
3	Modélisation du système	9
4	Implémentation du modèle	10
5	Compte-rendu de V & V	12
6	Résultats de la simulation	13
6.1	Retard en fonction du nombre de portes	13
6.1.1	Aéroport à 4 portes d'embarquement	14
6.1.2	Aéroport à 6 portes d'embarquement	14
6.1.3	Aéroport à 8 portes d'embarquement	15
6.1.4	Aéroport à 12 portes d'embarquement	16
6.2	Répartition journalière des arrivées	17
7	Analyse des résultats	19
7.1	Un aéroport saturé	19
7.2	Des solutions au problème	19
8	Perspectives d'évolutions	21
	Annexes	21
	Index	22
	Glossaire	22

Chapitre 1

Objectifs de l'étude

L'objectif de la présente étude est de simuler l'aéroport de Brest afin de dimensionner de façon optimale ses futures évolutions.

Pour cela nous avons réalisé une simulation à événement discret, dans laquelle nous avons étudié l'état de l'aéroport à chaque événement.

L'aéroport possède un seul terminal avec 4, 6 ou 8 emplacements ; une seule piste (6L/24R) et deux taxiways entre l'aérogare et la piste, un pour l'atterrissage et l'autre pour le décollage. Cette configuration est représentée sur la figure 1.1.

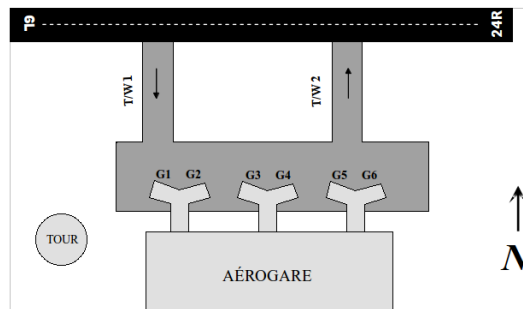


FIGURE 1.1 – Boucle de simulation

La simulation porte une attention particulière sur les arrivées des avions et les temps qu'ils mettent à effectuer les différentes étapes, de leur arrivée dans l'espace aérien de l'aéroport jusqu'à leur départ. La simulation a pour objectifs de déterminer les surcharges éventuelles de l'aéroport et leur contexte, en fonction des infrastructures disponibles. Elle est donc modulable, afin de faire varier ces infrastructures, notamment en modifiant les nombres de portes d'accès et de pistes, et d'étudier les conséquences de ces variations.

Le livrable final est composé du programme de simulation (sous la forme d'un projet JAVA compilable) et d'une étude des données générées par ce programme. Cette étude déterminera notamment les moyennes de fréquentation journalière, de retards, de temps d'attentes avant atterrissage.

Chapitre 2

Analyse du problème

Le système que nous allons simuler est un aéroport. Toutefois, notre modèle peut-être grandement simplifié par rapport à un aéroport réaliste. Puisque l'étude se focalise sur la surcharge du terminal, seul le processus d'atterrissage, de décollage et de traitement des avions nous intéresse. On peut donc ignorer de nombreux aspects, tels que la gestion des personnels, l'entretien des infrastructures, la gestion financière. La seule chose pertinente dans notre cadre est le statut des avions et des infrastructures (occupés ou non) au cours du temps. c'est à partir de ces informations que nous pourrions extraire les statistiques de fréquentation demandées. Ainsi les systèmes environnant en interaction avec le système étudié sont essentiellement constitué des Avions.

Les entités de simulations identifiées à la lecture du sujet sont :

- L'aéroport de Brest
- Les avions
- les infrastructures de l'aéroport, a savoir :
 - la piste
 - un taxiway utilisé à l'atterrissage
 - plusieurs portes (4, 6 ou 8)
 - un taxiway utilisé au décollage

Le cycle de vie des infrastructures est bien sur inclus dans celui de l'aéroport. C'est l'aéroport qui donne créé ses infrastructures. Nous allons dans les paragraphes qui suivent analyser chaque entité et identifier puis caractériser :

- leurs variables d'états
- leurs propriétaires d'initialisation et caractéristiques
- les événements temporels, et d'états
- les comportements et hypothèses de modélisation

2.1 L'aéroport

Entité	Aéroport
Type d'Entité	Permanente
Variables d'états	meteo waitingListLanding waitingListGate waitingListTW2 waitingListTakeOff
Variables statistiques de scrutation	
Paramètres techniques et données d'initialisation	currDate startDate stopDate evtinit nb_gate
Événements	EventsInitializer EvtRelease_P_TW1 EvtReleaseGate EvtRelease_TW2 EvtRelease_P
Comportement	
Processus Stochastique	Aucun

TABLE 2.1 – Analyse de l'entité aéroport

2.2 Les avions

2.3 Les infrastructures

2.4 Le scénario

2.5 Les listes d'attente

Entité	Avion
Type d'Entité	Temporaire
Variables d'états	status
Variables statistiques de scrutation	
Paramètres techniques et données d'initialisation	ID type
Événements	EvtOncomingPlane EvtApproach EvtLanding EvtRollIn EvtLoading EvtRefueling EvtRollOut EvtUnload EvtTakeOff
Comportement	
Processus Stochastique	Aucun

TABLE 2.2 – Analyse de l'entité avion

Entité	Aéroport
Type d'Entité	Permanente
Variables d'états	LB Michael Duberry Dominic Matteo Didier Domi
Variables statistiques de scrutation	MC Eirik Bakke Jody Morris
Paramètres techniques et données d'initialisation	MC Eirik Bakke Jody Morris
Événements	Alan Smith ST
Comportement	MC Eirik Bakke Jody Morris
Processus Stochastique	Aucun

TABLE 2.3 – Analyse de l'entité infrastructure

Entité	Aéroport
Type d'Entité	Permanente
Variables d'états	LB Michael Duberry Dominic Matteo Didier Domi
Variables statistiques de scrutation	MC Eirik Bakke Jody Morris
Paramètres techniques et données d'initialisation	MC Eirik Bakke Jody Morris
Événements	Alan Smith ST
Comportement	MC Eirik Bakke Jody Morris
Processus Stochastique	Aucun

TABLE 2.4 – Analyse de l'entité aéroport

Chapitre 3

Modélisation du système

Le système que nous simulons est composé de divers objets.

Les entités représentent les acteurs de la simulation. Cela inclut les lieux. L'aéroport représente l'ensemble de la simulation. Il contient une piste, deux taxiways, plusieurs portes d'embarquement. Il reçoit des avions, qui sont générés en grand nombre au cours de la simulation.

Les variables de la simulation sont moins nombreuses : La date permet de tenir compte du temps dans notre simulation. La météo (qui est soit bonne soit mauvaise) modifie le temps d'approche.

Les événements sont le cœur de la simulation. Ils sont déclenchés séquentiellement par la boucle de simulation. Ils sont tous datés. Ils planifient d'autres événements, modifient les entités et les variables et renvoient les informations qui serviront à l'analyse des résultats. L'événement de notification crée l'avion et le met en attente si la piste et le taxiway 1 ne sont pas disponibles. L'événement d'approche rend indisponible la piste et le taxiway 1. Il dure 2 à 5 minutes, doublé en cas de mauvais temps. L'événement d'atterrissage dure 2 minutes. Dès qu'il est fini, l'événement de relachement de piste rend celle-ci disponible. L'événement de roulage entrant occupe le taxiway 1 et dure 2 à 6 minutes. Dès qu'il est terminé, le taxiway 1 est libéré par l'événement correspondant si une porte est libre. Les événements de déchargement, de ravitaillement et d'embarquement durent respectivement 10, 30 et 20 minutes. À l'issue de ceux-ci, si le taxiway 2 est libre, l'événement de roulage sortant est déclenché et l'avion libère la porte pour occuper le taxiway 2. Le roulage dure là encore 2 à 6 minutes. Si la piste est libre, l'événement de décollage est déclenché. Il dure 3 minutes, à l'issue desquelles la piste devient à nouveau libre.

L'événement de changement de météo existe en parallèle des autres, il agit sur la variable de météo pour la passer de bonne à mauvaise et inversement.

Chapitre 4

Implémentation du modèle

Le système est composé de deux parties : un programme, écrit en Java, qui implémente la boucle de simulation, et un fichier Excel qui contient les données issues du programme et qui les traite pour en ressortir des résultats statistiques. Le programme Java implémente une boucle de simulation événementielle, telle que décrite par la figure 4.1, dans la classe principale Aéroport. Le processus d'initialisation remplit un agenda (de type `<Events>SortedList`) qui contient les événements d'arrivées d'avion dans l'espace aérien de l'aéroport, ainsi que les changements de météo. Ces événements sont générés par une boucle. Les intervalles de temps entre les événements dépendent de la date de l'événement précédent. Ceci est fait pour respecter le cahier des charges. Nous avons fait le choix de générer tous les événements d'arrivée d'avion et de changement de météo avant le parcours de la boucle principale de simulation. L'initialisation instancie également les objets représentant les infrastructures (tous de type `Facilities`), les avions étant générés durant la simulation par le traitement de l'événement correspondant à leur arrivée. Tous les événements sont des sous-classes de la classe abstraite `Events`. Chacun contient sa propre méthode *doSomething()* qui est appelée par la boucle de simulation. Cette méthode crée les événements dont découle l'événement courant et renvoie une chaîne de caractère décrivant l'événement qui sera intégrée dans le logger, le fichier retourné par le programme. L'architecture du programme Java est décrite dans le diagramme de classe 4.2.

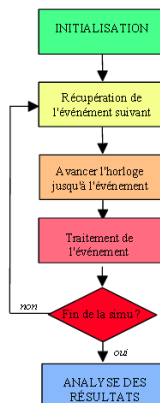


FIGURE 4.1 – Boucle de simulation

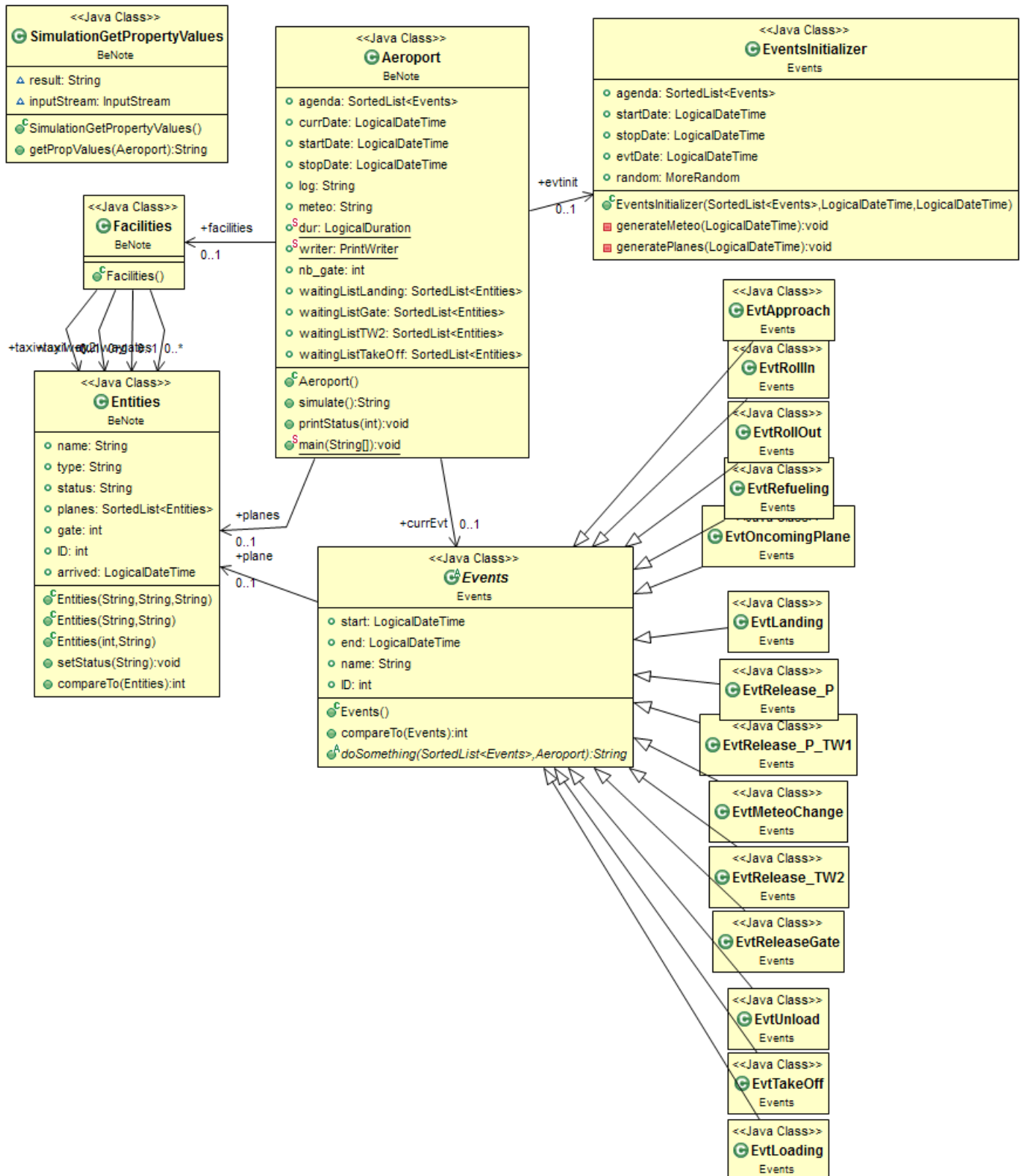


FIGURE 4.2 – Diagramme de classe

Chapitre 5

Compte-rendu de V & V

Revue critique du fonctionnement et des résultats.

Chapitre 6

Résultats de la simulation

La simulation a été exécutée sur 4 scénarios. Chaque scénario durait 90 jours, et voyait des avions arriver avec des probabilités identiques :

- Un avion toutes les 20 minutes en moyenne en horaire normal.
- Un avion toutes les 10 minutes en heure de pointe : de 7h à 10h et de 17h à 19h, en semaine.
- Un avion toutes les 40 minutes le week-end.
- Aucun avion de 22h à 7h.

Toutes les avions ont le même comportement, tel que décrit dans la modélisation du problème.

Les quatre scénarios ont pour seule différences le nombre de portes d'embarquement instanciées : 4, 6, 8 ou 12.

On étudie les horaires de notification des avions à la tour de contrôle et de début de phase d'approche. La différence entre ces deux dates indique le retard de l'avion, c'est à dire le temps que l'avion a passé en l'air à attendre la permission d'atterrir. Les moyennes de ces retards sont compilés dans le tableau 6.1

4 portes	6 portes	8 portes	12 portes
123 :12 :35	544 :06 :03	356 :29 :14	0 :08 :10

TABLE 6.1 – Retard moyen

6.1 Retard en fonction du nombre de portes

Voici ci dessous des répartitions plus détaillées des retards en fonction du nombre de portes :

6.1.1 Aéroport à 4 portes d'embarquement

La répartition des retards pour un aéroport à 4 portes, en pourcentage d'avions dans des intervalles de retards est donnée en figure 6.1.

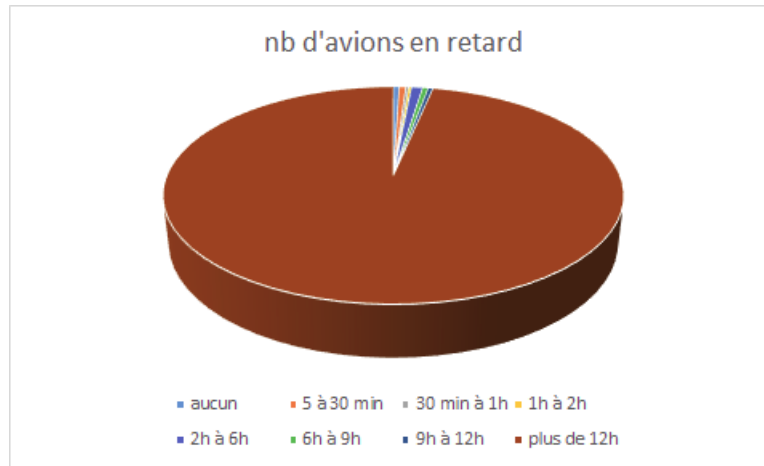


FIGURE 6.1 – Répartition des retards, avec 4 portes

Une autre donnée intéressante est la répartition des retards par jour, donnée par la figure 6.2

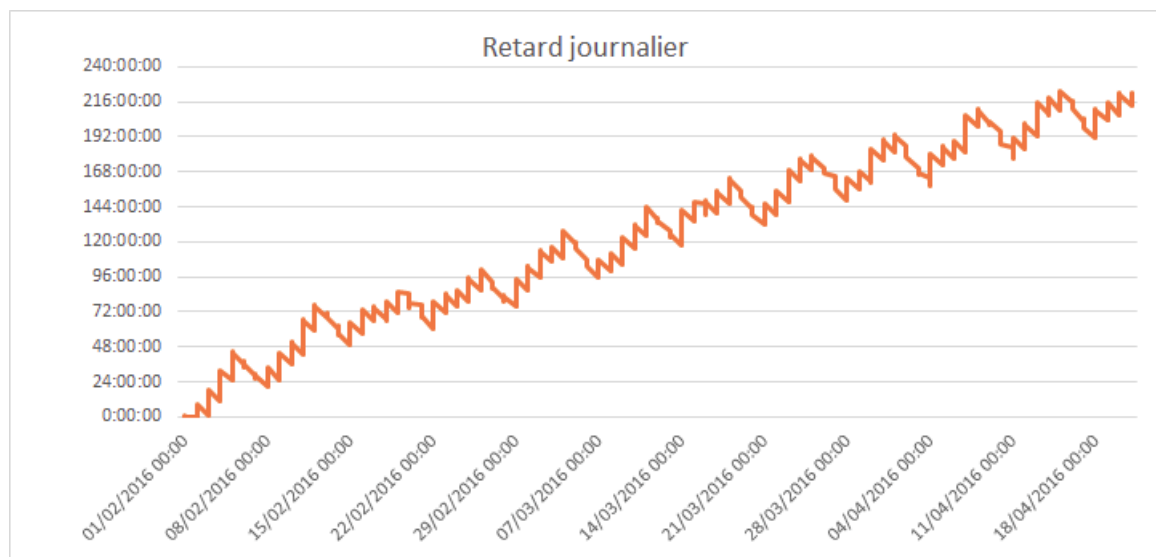


FIGURE 6.2 – Retard avec 4 portes

6.1.2 Aéroport à 6 portes d'embarquement

La répartition des retards pour un aéroport à 6 portes, en pourcentage d'avions dans des intervalles de retards est donnée en figure 6.3.

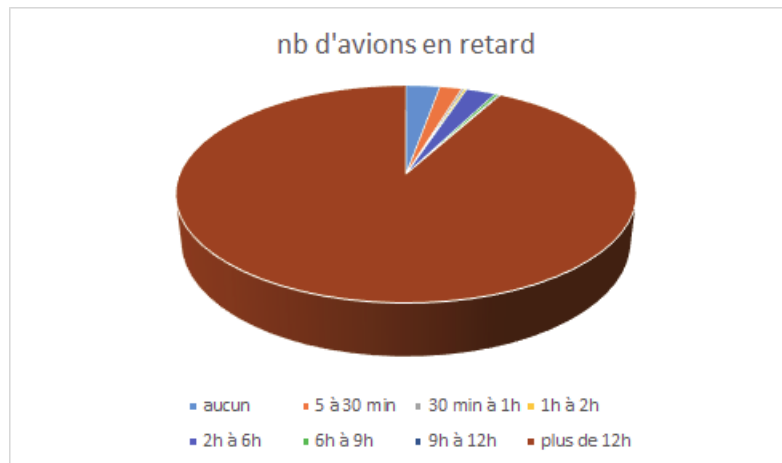


FIGURE 6.3 – Répartition des retards, avec 6 portes

Une autre donnée intéressante est la répartition des retards par jour, donnée par la figure 6.4

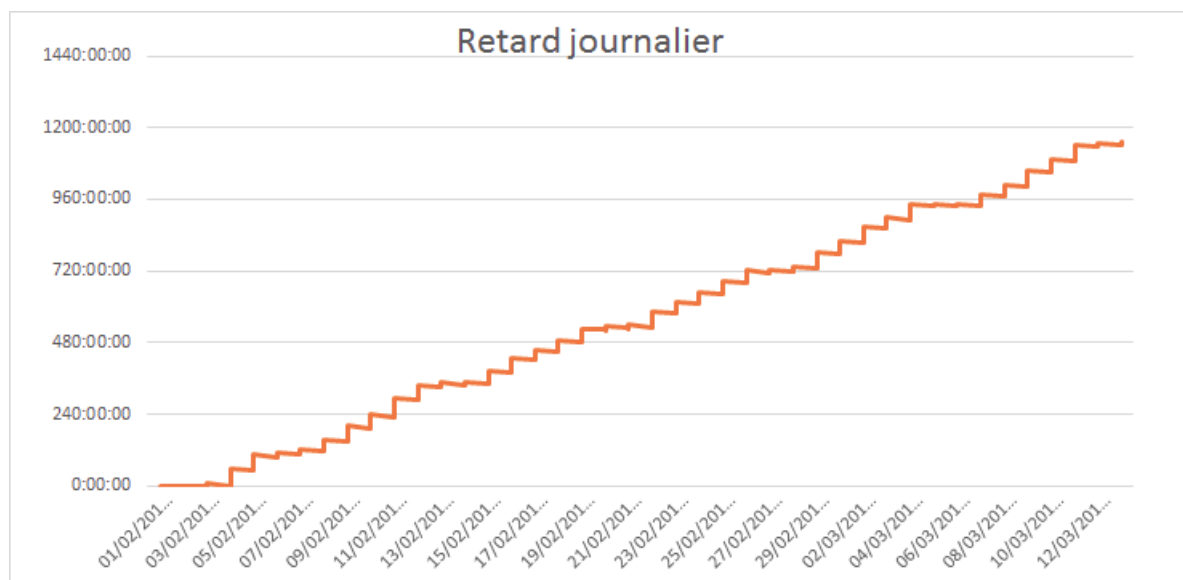


FIGURE 6.4 – Retard avec 6 portes

6.1.3 Aéroport à 8 portes d'embarquement

La répartition des retards pour un aéroport à 8 portes, en pourcentage d'avions dans des intervalles de retards est donnée en figure 6.5.

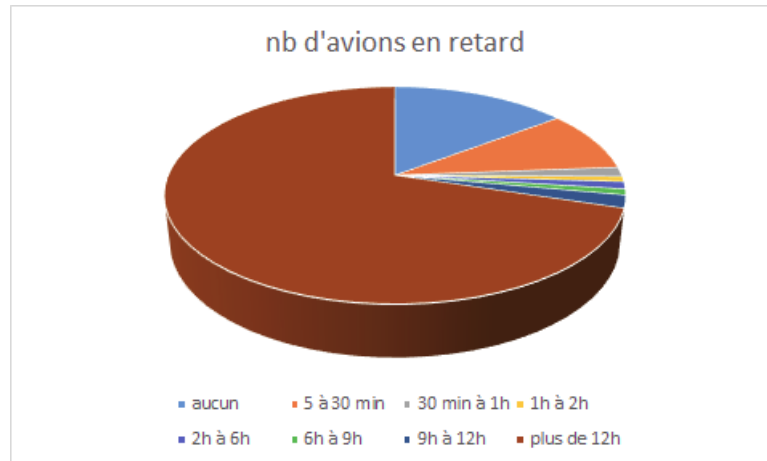


FIGURE 6.5 – Répartition des retards, avec 8 portes

Une autre donnée intéressante est la répartition des retards par jour, donnée par la figure 6.6

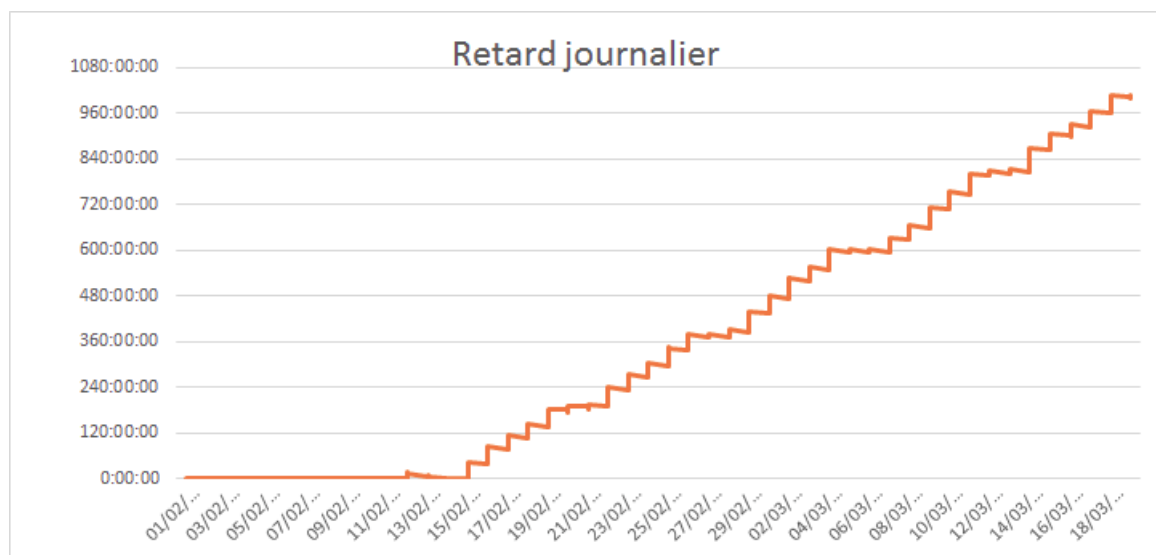


FIGURE 6.6 – Retard avec 8 portes

6.1.4 Aéroport à 12 portes d'embarquement

La répartition des retards pour un aéroport à 12 portes, en pourcentage d'avions dans des intervalles de retards est donnée en figure 6.7.

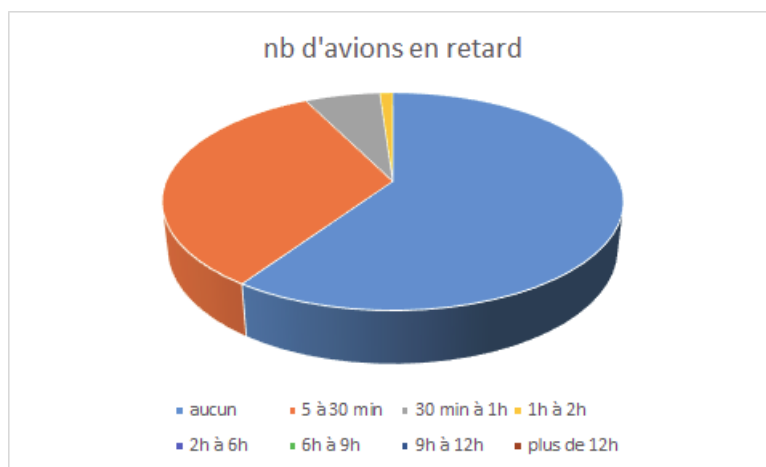


FIGURE 6.7 – Répartition des retards, avec 12 portes

Une autre donnée intéressante est la répartition des retards par jour, donnée par la figure 6.8

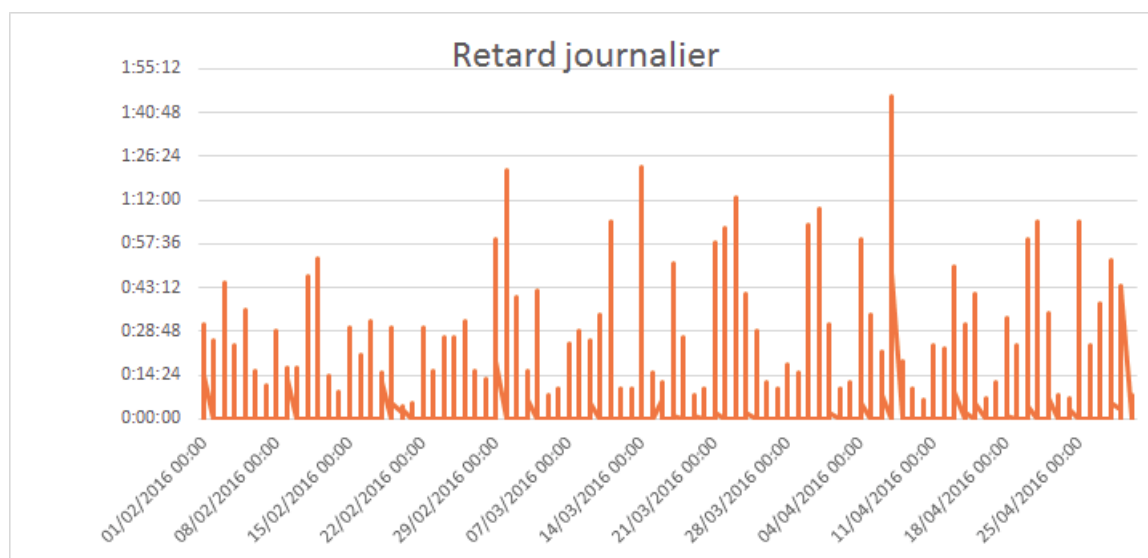


FIGURE 6.8 – Retard avec 12 portes

6.2 Répartition journalière des arrivées

Quelque soit le nombre de portes, les nombres moyens d'atterrissage par heure –respectivement en jour de semaine et en week-end– restent les mêmes, tels que donnés dans les figures 6.9 et 6.10.

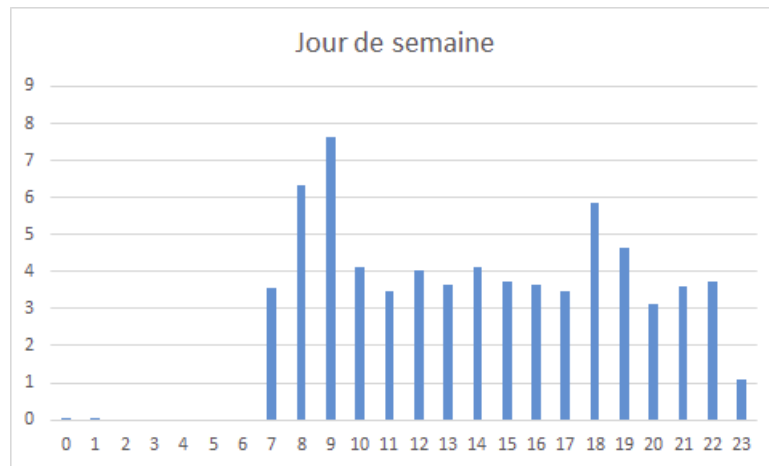


FIGURE 6.9 – Quantité d’atterrissage par horaires en semaine

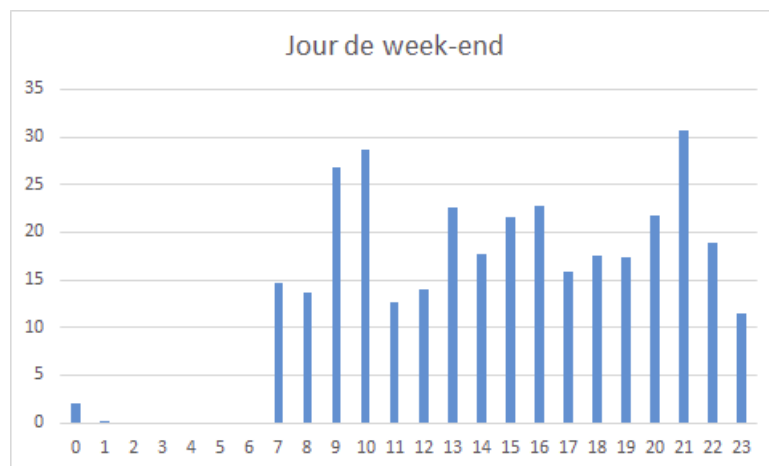


FIGURE 6.10 – Quantité d’atterrissage par horaires le week-end

Chapitre 7

Analyse des résultats

7.1 Un aéroport saturé

Au vu des résultats précédents, il semble évident que l'aéroport tel qu'il est simulé dans les cas à 4, 6 et 8 portes (nous reviendrons sur le cas à 12 portes plus loin) ne fonctionne pas de façon optimal. Les retards s'accumulent très vite, d'autant plus vite que le nombre de portes est faible. Puisque le système de l'aéroport n'est pas capable de gérer les avions plus vite qu'ils n'arrivent, ceux-ci s'accumulent, tout d'abord dans les terminaux puis dans la liste d'attente pour l'atterrissage. Lorsque la nuit tombe et que les avions cessent d'arriver à l'aéroport, la liste commence à se vider, tous les avions en attente se posant sans être remplacés. On tombe alors dans deux cas : soit la liste est suffisamment courte pour que tous les avions se posent avant la réouverture de l'aéroport, auquel cas le lendemain l'aéroport peut reprendre son activité telle quelle, soit la nuit entière ne suffit pas à vider la file, auquel cas les premiers avions arrivant au matin ne peuvent pas atterrir et passent dans la file d'attente qui commence à grandir à chaque jour, entraînant des retards de plus en plus longs. Les week-ends, avec leur fréquentation moindre, peuvent aider à désengorger le flux d'avions en attente mais ne sont en général pas suffisants. Une bonne illustration de ce phénomène est la figure 6.6. En effet, jusqu'au 15/02, les retards restent faibles : on observe des pics mais ils sont résorbés avant la fin de la journée. Toutefois, à partir du 15/02, le flux devient ingérable et les retards augmentent de façon linéaire avec le nombre de jours. Le moment où l'aéroport passe en mode saturé advient d'autant plus tôt que le nombre de portes est faible.

Le cas de l'aéroport à 12 portes est particulier et peut apporter un début de réponse. L'aéroport ne passe pas en mode saturé au cours des 90 jours de simulation. Il semble que le rôle de tampon joué par les portes d'embarquement permet de fluidifier suffisamment le trafic pour qu'il n'y ait jamais de blocage et donc de saturation.

7.2 Des solutions au problème

Nous avons vu à la section précédente qu'un plus grand nombre de portes d'embarquement permet de fluidifier le système. La solution la plus évidente consiste

donc à construire plus de portes d'embarquement. Toutefois d'autres solutions peuvent être envisagées. Bien que non testée dans notre simulation, il nous semble important d'en faire part ici, afin que des études puissent être menées dans ces directions. Une des solutions alternatives face à un trafic trop important est de réduire ce trafic. Réduire le nombre d'avions arrivant à l'aéroport permettrait de mieux traiter chaque avion. Nous sommes bien conscient que la tendance actuelle n'est pas au ralentissement du trafic aérien, mais il nous apparaissait important de l'évoquer. Une autre solution serait l'ajout d'une seconde piste d'atterrissage. Une piste supplémentaire permettrait en effet plus de souplesse dans la gestion des flux.

Chapitre 8

Perspectives d'évolutions

Suggestions d'amélioration du logiciel de simulation, du modèle...

