



Laboratoire IBISC
40, Rue du Pelvoux
91020 Évry



Université d'Évry Val-d'Essonne
23, Boulevard François Mitterrand
91000 Évry

RAPPORT DE STAGE

MASTER 2 E3A - RVSI

Processus de difficulté adaptative pour la rééducation motrice post-AVC du membre supérieur

Étudiant :
M. David REI
Promotion 2018

Tuteur :
M. Guillaume BOUYER
Laboratoire IBISC

Mars 2018 - Septembre 2018

Table des matières

1	Introduction	5
1.1	Contexte	5
1.1.1	Laboratoire	5
1.1.2	Projet	6
1.2	Problématique détaillée	8
2	Etat de l'art	10
2.1	Définition	10
2.2	Méthodes par seuils	11
2.3	Méthodes par apprentissage	14
2.4	Méthodes prédictives et probabilistes	18
2.5	Méthodes par scores	23
2.6	Bilan	26
3	Modèle de description et d'adaptation de la difficulté	29
3.1	Introduction	29
3.2	Exercices, aptitudes et autres définitions	30
3.2.1	Exercice	31
3.2.2	Aptitudes	31
3.2.3	Transitions	32
3.2.4	Configurations	33
3.3	Modélisation de la difficulté d'un exercice	33
3.3.1	Calibration	33
3.3.2	Principe	34
3.3.3	Difficulté d'aptitudes	34
3.3.4	Difficulté de l'endurance	36
3.3.5	Difficulté globale	39
3.3.6	Conclusion	39
3.4	Génération d'exercices	40
3.4.1	Problématiques	40
3.4.2	Génération par la difficulté	41
3.4.3	Exemples de génération contrainte	43
3.5	Évaluation et Adaptation	46

3.5.1	Évaluation de la performance	46
3.5.2	Adaptation	47
3.6	Conclusion	48
4	Conclusion et perspectives	50
4.1	Conclusion du stage	50
4.2	Conclusion personnelle	51
4.3	Perspectives	51

Remerciements

En 2015, j'ai découvert à Prague, pendant mon stage, deux domaines qui me passionnent aujourd'hui : la recherche et la réalité virtuelle (RV). Ces deux domaines sont responsables de ma présence ici, dans ce master 2 Recherche. Très rapidement dès mon arrivée, j'ai manifesté mon intérêt pour la thèse de doctorat, après le master, sous réserve de réussite dans le parcours.

Monsieur Guillaume Bouyer a remarqué cette motivation, et dès qu'il en a eu l'occasion, il m'a proposé un stage avec lui, dans l'équipe IRA2 du laboratoire IBISC, avec à son issue une poursuite possible en thèse de doctorat. Je le remercie de m'avoir donné cette opportunité de travailler avec lui et d'avancer sur l'une des nombreuses problématiques du laboratoire.

Je remercie également Monsieur Ludovic David, actuellement doctorant dans cette même équipe. Nous avons collaborés dans le cadre du projet "Cesaar-AVC" aux côtés de l'entreprise "*Interaction Healthcare*". J'ai beaucoup apprécié ton aide, et surtout nos nombreuses réunions et discussions autour du projet, de mon stage, et de tout le reste.

Enfin, je remercie toute l'équipe qui m'a accueilli au laboratoire, notamment Mesdames Aylene Ricca et Mouna Takherist, et Monsieur Amine Chellali. Le stage s'est déroulé dans un cadre chaleureux, détendu et agréable.

Résumé

L'Accident Vasculaire Cérébral (AVC) est une maladie qui affecte plus de 150 000 patients par an en France, dont environ la moitié garde des séquelles fonctionnelles. Le déficit moteur du membre supérieur en est une des principales. La rééducation motrice est une méthode qui permet d'intensifier la récupération, celle-ci restant de qualité variable.

L'apport des technologies de la Réalité Virtuelle (RV) à la rééducation motrice n'est aujourd'hui pas encore totalement validé par l'ensemble des organismes de santé. Cependant, une très forte tendance émerge et montre des résultats positifs et encourageants. En collaboration avec la clinique "*Les Trois Soleils*" et la société "*Interaction Healthcare*", le projet "*Cesaar-AVC*" vise à concevoir et tester un premier prototype d'outil interactif d'aide à la rééducation motrice du membre supérieur. Il s'agit d'un jeu thérapeutique de pronation-supination de la main qui consiste à contrôler le déplacement d'un personnage par un mouvement de rotation de la main afin de ramasser des objets dans un environnement virtuel immersif. Le suivi de la main réelle du patient est réalisé à travers un dispositif de capture bas coût et la visualisation à travers un écran multimédia.

Actuellement, ce jeu thérapeutique ne possède qu'un seul mode de difficulté. Le but de ce projet est donc de concevoir, développer puis tester un processus d'adaptation de la difficulté permettant de personnaliser l'expérience du patient à ses performances, étaillonnées au préalable par une phase de calibration.

La solution proposée s'articule autour de plusieurs segments : un modèle de description de la difficulté, une phase de calibration du système, permettant son initialisation avec les performances du patient, et une session, pendant laquelle le patient effectue ses exercices, et où plusieurs événements s'enchaînent dans l'ordre suivant. Le système génère tout d'abord les exercices, en fonction des contraintes de génération définies par le modèle ainsi que des résultats de l'exercice précédent ; le patient complète ensuite les exercices proposés, ses performances sont mesurées en temps réel, tout au long de sa tâche. Enfin, l'évaluation des performances puis la décision de l'adaptation sont effectuées, afin d'apporter les informations les plus pertinentes possibles pour la génération de l'exercice suivant.

Au moment de l'écriture de ce rapport, la solution est en cours d'implémentation dans le jeu thérapeutique de l'oiseau. Certains points restent à finaliser, notamment l'ajustement des coefficients des aptitudes qui contrôlent le comportement du modèle de description de la difficulté.

Chapitre 1

Introduction

1.1 Contexte

1.1.1 Laboratoire

Le laboratoire IBISC (Informatique, Biologie Intégrative et Systèmes Complexes), intégré à l'Université d'Evry Val-d'Essonne, articule son projet scientifique autour de deux axes :

- L'axe STIC et Vivant, qui développe des compétences pour la bioinformatique et l'étude des systèmes biologiques d'une part ; et d'autre part, conçoit, évalue et réalise des systèmes d'assistance à la personne dans son environnement, intérieur ou extérieur.
- L'axe STIC et Systèmes Intelligents dont l'objet est d'étudier et de concevoir des systèmes (au sens large) autonomes et intelligents, et aussi concevoir de manière maîtrisée des systèmes ouverts, i.e. qui doivent être capables de s'adapter à n'importe quelle modification de leur environnement et de trouver un comportement adéquat.

Pour mener à bien ses missions dans ces deux axes de recherche, le laboratoire IBISC est constitué de quatre équipes :

- AROBAS (Algorithmique, Recherche Opérationnelle, Bioinformatique et Apprentissage Statistique) : recherche fondamentale et appliquée en algorithmique et en apprentissage automatique, notamment pour des applications de bioinformatique.
- COSMO (COmmunications, Spécifications MOdèles) : étude des propriétés fondamentales de systèmes dynamiques réactifs, décentralisés et ouverts, ainsi que de leur comportement.
- IRA2 (Interactions, Réalité Augmentée, Robotique Ambiante) : conception, réalisation et évaluation de systèmes d'assistance à la personne par l'utilisation et la

maîtrise de systèmes complexes artificiels. Il s'agit de l'équipe qui m'a accueilli pour mon stage.

- SIMOB (Systèmes Intelligents, MOdélisation et Biologie) : étude de systèmes tels que les véhicules et les systèmes biologiques, par l'étude générale en quatre étapes que sont la perception, l'observation, la modélisation et la commande de tels systèmes.

Équipe IRA2

Deux plateformes sont à disposition pour mettre en oeuvre et évaluer les travaux de recherche de l'équipe : EVR@ et ARPH. Ces travaux sont organisés autour de trois objectifs :

- objectif 1 : proposer des assistances multisensorielles et collaboratives, pour l'interaction et la prise de décision en situation d'immersion ou de semi immersion dans un contexte de réalité virtuelle.
- objectif 2 : proposer des assistances à la personne en interaction directe, dans un contexte de réalité augmentée.
- objectif 3 : proposer des fonctionnalités à un "robot assistant" en coopération avec un opérateur, évoluant dans un environnement de robotique ambiante.

Les champs d'application de ces objectifs sont très différents, concernant autant les personnes valides qu'en déficit d'autonomie. L'assistance envisagée par ces travaux peut être à domicile, en institution, en itinérance ou dans un milieu professionnel.

1.1.2 Projet

L'Accident Vasculaire Cérébral (AVC) est une maladie qui affecte plus de 150 000 patients par an en France, dont environ la moitié garde des séquelles fonctionnelles. Le déficit moteur du membre supérieur est une des principales. La rééducation motrice est une méthode qui permet d'intensifier la récupération, celle-ci restant de qualité variable. En unité de soin, cette rééducation est en général initiée après un délai moyen de 7 à 10 jours. Cependant, elle est limitée du fait de la relative disponibilité des kinésithérapeutes pour chaque patient et des faibles moyens disponibles pour l'éventuelle auto-rééducation. Après une dizaine de jours, le patient poursuit en général sa rééducation en centre spécialisé où les exercices thérapeutiques sont plus variés et mieux encadrés.

L'apport des technologies de la Réalité Virtuelle (RV) à la rééducation motrice n'est aujourd'hui pas encore totalement validé par l'ensemble des organismes de santé. Cependant, une très forte tendance émerge et montre des résultats positifs et encourageants. En collaboration avec la clinique "*Les Trois Soleils*" et la société "*Interaction Healthcare*",

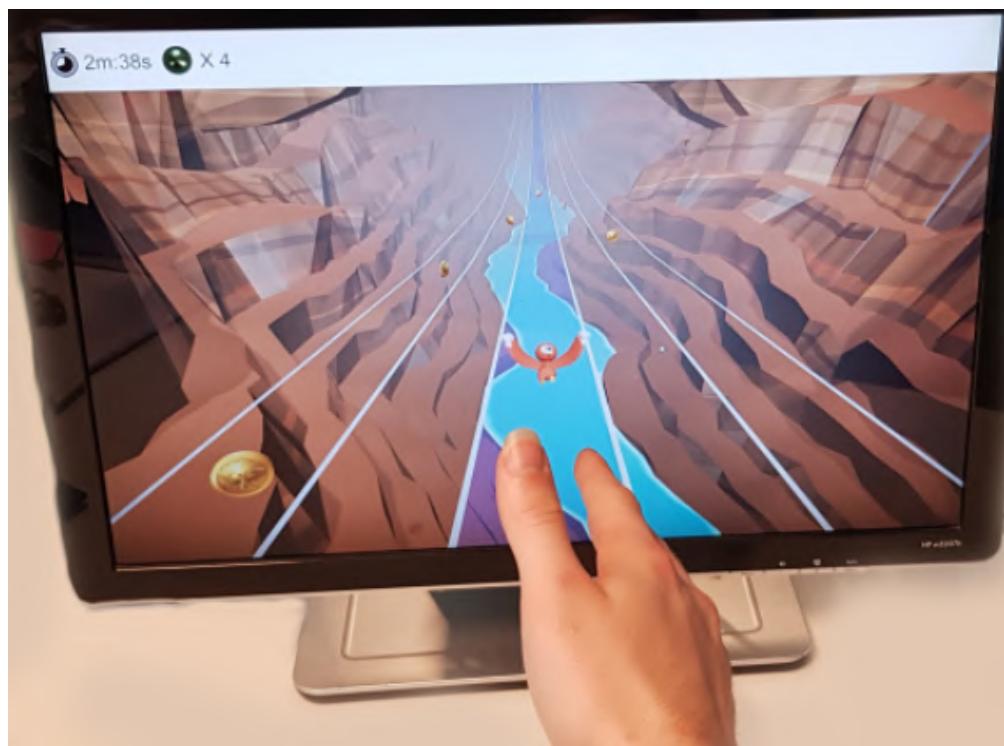


FIGURE 1.1 – Jeu thérapeutique de pronation-supination de la main (jeu de l'oiseau).

le projet "Cesaar-AVC" vise à concevoir et tester un premier prototype d'outil interactif d'aide à la rééducation motrice du membre supérieur. Dans ce projet, plusieurs jeux thérapeutiques sont en cours de développement. Celui qui fut le plus utilisée dans le cadre de mon stage est un jeu thérapeutique de pronation-supination de la main qui consiste à contrôler le déplacement d'un personnage (ici un oiseau) par un mouvement de rotation de la main afin de ramasser des objets (ici des pièces) dans un environnement virtuel immersif. De manière plus précise, le personnage avance verticalement dans la scène de manière automatique et le patient contrôle le mouvement horizontal (Fig. 1.1).

La scène est divisée en lignes et colonnes, formant une grille sur laquelle les pièces sont réparties de manière particulière, formant un chemin que le patient doit suivre afin de travailler quatre aptitudes du mouvement de pronation-supination : l'amplitude, le maintien, la vitesse et la précision.

Le modèle décrit est basé pour le moment sur une grille à 5 colonnes et 20 lignes comportant chacune 1 pièce (donc 20 pièces). La colonne 1 désigne la zone de départ, et de repos du patient. La colonne 5 désigne la colonne maximale, la plus éloignée.

Le suivi de la main réelle du patient est réalisé à travers un dispositif de capture bas coût (*Leap Motion*) et la visualisation à travers un écran multimédia (Fig. 1.2).

Actuellement, ce jeu thérapeutique ne possède qu'un seul mode de difficulté. Le but

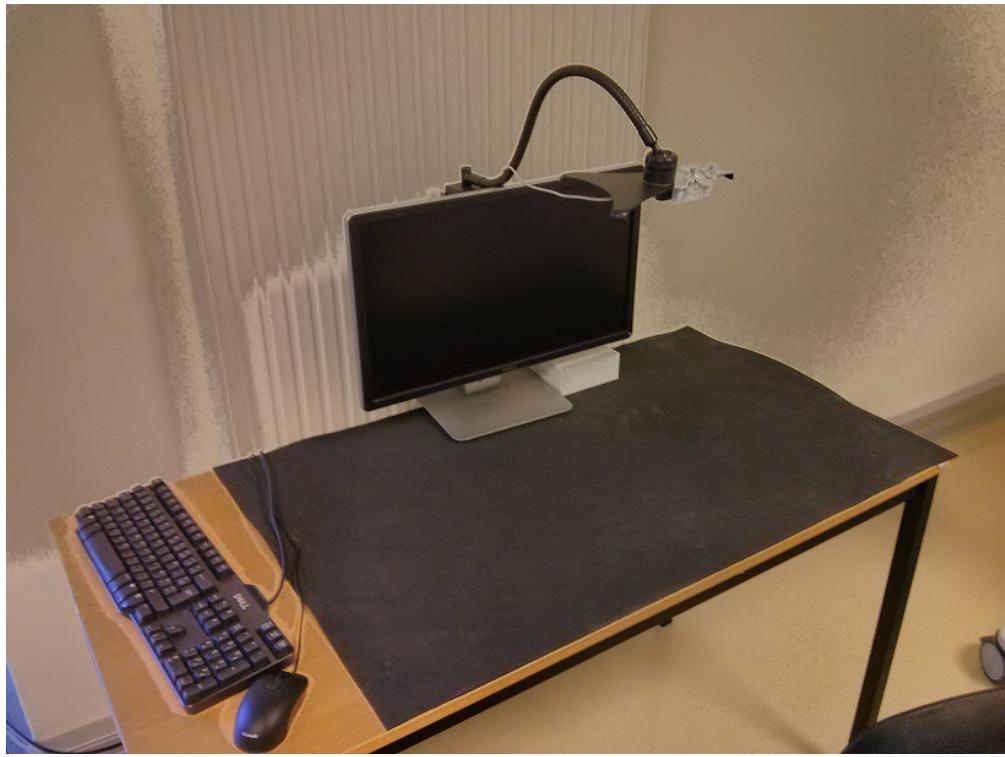


FIGURE 1.2 – Dispositif complet de capture et de visualisation

de ce projet est donc de concevoir, développer puis tester un processus d'adaptation de la difficulté permettant de personnaliser l'expérience du patient à ses performances.

1.2 Problématique détaillée

Le problème possède plusieurs facettes. Tout d'abord, les patients forment un ensemble très **hétérogène**, de personnes âgées de 18 à 99 ans, avec une moyenne d'âge de 70 ans, où chaque individu possède des **capacités uniques** qui ne peuvent être comparées à aucun autre patient. Après un accident vasculaire cérébral, certains patients n'ont que des incapacités motrices ; d'autres ont des incapacités cognitives associées. Par ailleurs, pour un même individu, les capacités motrices et/ou cognitives peuvent être changeantes **du jour au lendemain**, de manière positive ou négative, indépendamment de la volonté du patient. Cette particularité est très importante, notamment en ce qui concerne la présentation d'un exercice d'une difficulté donnée, qui potentiellement peut désarçonner le patient. De plus, la rééducation étant un processus à très long terme, elle sera amenée à évoluer, quoique la prédiction de son évolution soit compliquée par les éléments mentionnés ci-dessus.

Il s'agit donc de proposer des exercices qui entretiennent **la motivation et l'intérêt** de

chaque patient dans la rééducation. Pour ce faire, ce projet propose de s'intéresser à **la difficulté adaptative, qui peut se définir comme une épreuve capable d'évoluer, de s'adapter ou de se modifier, de manière autonome ou non**. Pour entretenir la motivation et l'intérêt des patients, il sera nécessaire de les observer, au travers de paramètres réels ou virtuels, puis de procéder à des modifications ayant une influence directe sur la motivation et l'intérêt.

Enfin, un tel système doit non seulement garder le patient motivé, mais il doit aussi tenir compte de tous les aspects de la rééducation, en particulier les aptitudes travaillées qui constituent les objectifs à long terme de celle-ci. La question est alors : **comment proposer aux patients des exercices adaptés à leurs capacités et adaptables à leurs évolutions tout au long du processus de rééducation ?**

Il est donc question dans un premier temps d'établir un état de l'art de la difficulté adaptative, en général et dans tous les domaines, puis de recentrer la recherche sur la rééducation motrice post-AVC, en particulier la rééducation du membre supérieur. De cette étude, nous extrairons une synthèse permettant de nous situer dans le paysage des processus de difficulté adaptative. Dans un second temps, et selon les apports de l'état de l'art, il sera question de concevoir, d'adapter ou d'implémenter un tel processus dans le "*jeu de l'oiseau*", le prototype actuellement développé en collaboration avec la clinique "*Les Trois Soleils*" et la société "*Interaction Healthcare*", dans le cadre du projet global "*Cesaar-AVC*". Enfin, un point sera fait sur l'état actuel du projet et les étapes qu'il reste à accomplir en vue de sa complétion.

Chapitre 2

Etat de l'art

2.1 Définition

Les jeux vidéo et par la suite les jeux thérapeutiques nécessitent de proposer un défi continu, afin d'entretenir l'intérêt du joueur pour les premiers, ou la motivation à suivre un programme de rééducation pour les seconds. Proposer un défi continu se révèle cependant difficile. Dans les jeux vidéo cela consiste habituellement à **proposer un réglage global de difficulté, puis des niveaux successifs de difficulté croissante**. Toutefois, il peut y avoir une grande variabilité entre les joueurs. Dans le cadre de la thérapie, cette variabilité inter-individuelle est complétée par une variabilité intra-individuelle, car les patients sont sujets à des changements d'état de forme ou de santé.

Un système de difficulté adaptative consiste en une ou plusieurs épreuves (ou défis) capables d'évoluer ou de se modifier en fonction des circonstances, afin d'accomplir un objectif déterminé et connu.

Ces systèmes existent depuis le début du millénaire. Ils furent tout d'abord utilisés dans les jeux vidéo, puis élargis à d'autres secteurs d'activités, tels que la rééducation (motricité, cognition, phobies...) par exemple. Un tel système repose sur quatre étapes successives permettant l'adaptation de la difficulté (Fig. 2.1). Chaque étape dépend du résultat de la précédente afin de se réaliser correctement.

- La phase de **calibration** permet d'initialiser le système, l'outil, avec les capacités initiales de l'utilisateur.
- Ensuite, la phase de **mesures** enregistre les performances de l'utilisateur pendant la réalisation de la tâche demandée. Cet enregistrement peut se faire sur de nombreux paramètres, à la fois des paramètres réels, tels que des mesures physiologiques (EEG, EMG, ECG [12], imagerie thermique du visage [7]) et des mesures physiques (amplitude de mouvement, temps de complétion de la tâche, rapport

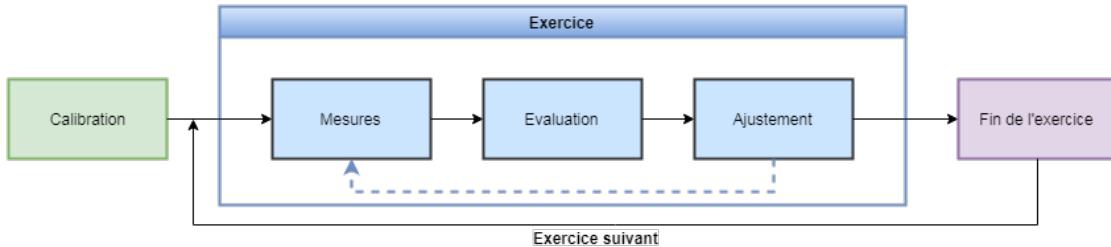


FIGURE 2.1 – Phases d'un système de difficulté adaptative.

entre vitesse moyenne et vitesse maximum [9]). Il peut également prendre en paramètre d'entrée des mesures virtuelles, telles que des informations extraites directement de la scène (position de l'utilisateur, progression générale, fréquence d'échec, nombre d'essais... [1]).

- La phase **d'évaluation** permet de juger des performances, en les comparant à une notation, un score ou un objectif fixé au préalable.
- Enfin, la phase **d'ajustement, ou d'adaptation**, permet de modifier la difficulté, en fonction du résultat de l'évaluation. La difficulté peut être une simple valeur à modifier, comme par exemple le temps imparti [2] ou l'emplacement d'une cible [10] ou alors un ensemble de paramètres ayant une influence directement sur le jeu ou les ressources du joueur [1][6][11]. La phase d'adaptation peut se produire pendant la session de jeu (intra) ou bien en dehors de la session (inter).

Le système de difficulté adaptative est l'ensemble de ces quatre phases. La mise en oeuvre de ces phases peut être faite selon plusieurs méthodes présentées dans les parties suivantes.

2.2 Méthodes par seuils

Les méthodes par seuils consistent à définir un seuil de performance de l'utilisateur à partir duquel changer le niveau de difficulté. Ce seuil peut être le rapport entre les succès et les échecs de la tâche demandée [2][3].

Ma et Bechkoum [2] ont par exemple développé un système d'aide à la rééducation post-AVC basé sur une thérapie par le mouvement et un serious game. Ce système se décompose en quatre segments (Fig. 2.2), auxquels s'ajoutent des briques fonctionnelles particulières.

Les quatre segments sont les suivants : les dispositifs d'entrée, tels les gants de données, le clavier ou la souris ; le moteur graphique, comprenant les modèles 3D ; la thérapie de mouvement, avec pour composantes les tâches fonctionnelles et les serious games, auxquelles s'attachent l'adaptation dynamique, les profils patients et l'analyse des résul-

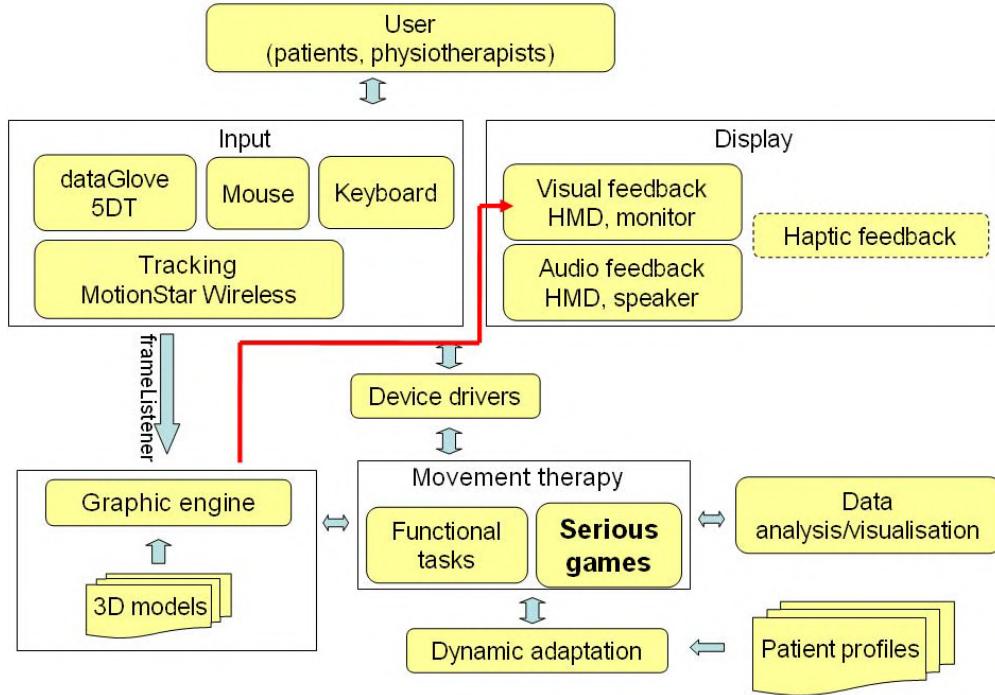


FIGURE 2.2 – Système multi-segments de Ma et Bechkoum[2].

tats ; et enfin le segment d'affichage : feedbacks visuels, audios et haptiques.

Ma et Bechkoum ont développé trois serious games utilisant les mouvements du membre supérieur comme contrôles. La configuration initiale du jeu est réalisée par rapport au profil patient : un temps plus important est accordé pour les déficiences motrices plus prononcées. Ces réglages sont modifiés dynamiquement pendant le jeu, pour s'adapter aux performances du patient. Trois niveaux de difficulté préétablis permutent en fonction de la réussite ou de l'échec du patient. Dans un même niveau de difficulté, certains éléments tels que le temps imparti à la réalisation de la tâche peuvent également varier. Certains éléments comme le score, le niveau et la vitesse adaptée apparaissent à l'écran comme feedbacks visuels, tandis que d'autres éléments tels que les données de performance ne s'affichent qu'en fin de partie. Les données de fin de partie seront ensuite utilisées comme données de configuration pour la partie suivante.

D'autres feedbacks sont affichés : changement de couleur de certains items cibles, avec un son propre à cet item, tableaux de statistiques, meilleurs scores personnels.

Un autre exemple de seuil simple est les travaux d'Alankus et al. [3], qui présentent d'un point de vue thérapeutique une méthode de conception de jeux de rééducation à domicile. L'aspect adaptatif de leur méthode réside dans l'établissement de modèle de joueurs, notamment en mesurant l'amplitude maximale des mouvements, où pour chaque

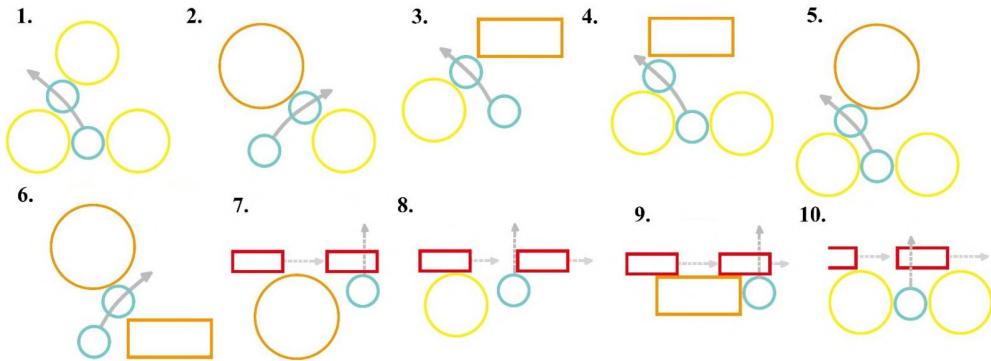


FIGURE 2.3 – Patterns de types d’obstacles de Li et al. [8]

patient correspond un modèle. Ensuite, ce sont les thérapeutes qui définissent la difficulté en créant des profils de difficulté personnalisés à chaque patient, et éventuellement en visant certains objectifs précis (précision, amplitude de mouvement, etc.). Dans ce système, la difficulté est adaptée de manière dynamique pour certains jeux, avec des seuils de déclenchement en fonction des succès et échecs de chaque patient.

Le seuil peut aussi être sous la forme de jauge comptabilisant les succès et les échecs indépendamment l’un de l’autre.

Li et al. [8] utilisent ces jauge pour un système de difficulté adaptative dans un jeu de plateformes, avec de la génération procédurale contrainte. Le système d’adaptation se déclenche lorsque l’utilisateur a passé 15 obstacles (augmentation de la difficulté puis ajustement des paramètres) ou subit 3 échecs dans le même niveau (diminution de la difficulté).

Les obstacles sont introduits sur une ligne horizontale, de manière aléatoire contrainte. En effet, il existe trois types d’obstacles (couteaux, bombes, tirs laser) qui ont chacun leurs propres dimensions. La contrainte réside dans l’impossibilité d’avoir deux obstacles identiques qui se suivent : il y a alors dix combinaisons possibles de deux types d’obstacles (Fig. 2.3), où les cercles verts représentent les mouvements du personnage, les autres formes étant les obstacles. L’intervalle minimal l entre deux obstacles est calculé de plusieurs manières, selon que les obstacles sont des cercles ou des rectangles. Il s’agit de l’un des paramètres d’adaptation de la difficulté, puisque l’intervalle diminue lorsque la difficulté augmente. En d’autres termes, le jeu devient plus difficile car l’espace alloué à l’utilisateur pour passer les obstacles est réduit.

Les bonus ne sont introduits qu’après avoir passé les 30 premiers obstacles, et apparaissent au rythme d’un tous les 2 écrans défilés. Ils sont de deux types : pièces d’or, qui augmentent le score du joueur, et “pain blanc” qui permettent de continuer le jeu en cas

de défaite.

La difficulté est calculée comme une fonction de la vitesse de chute des obstacles, de l'intervalle de temps entre deux obstacles tombants, de la proportion des 10 plus difficiles combinaisons parmi toutes les combinaisons apparues, et de la fréquence des situations trompeuses (situations où une surenchère de bonus amène à un échec).

Les autres paramètres ajustés sont les règles d'apparition des bonus, qui seront ajoutés dans des situations à la criticité plus ou moins élevée.

L'étude préliminaire a montré que ce système peut s'adapter à des joueurs de niveaux très différents (de débutant à expert), et proposer dans tous les cas un défi correspondant à leurs propres performances.

Les méthodes par seuils constituent une première approche d'un système d'adaptation de la difficulté. Elles consistent donc à prendre un seuil comme référence du déclenchement de l'adaptation, qui peut à son tour ajuster un ou plusieurs paramètres.

2.3 Méthodes par apprentissage

Les méthodes par apprentissage consistent à définir un modèle sur lequel le système va s'entraîner en amont, afin de reconnaître les situations nécessitant une adaptation. Ces méthodes peuvent être mises en place sur une base d'utilisateurs très importante, telle que celle utilisée par Jennings-Teats et al. [4] par exemple, ou sur des caractéristiques propres à chaque utilisateur telles que celles employées par Hocine et al. [10].

Le système proposé par Jennings-Teats et al. concerne les jeux de plateformes. Il est question de construire un niveau de jeu par la génération de briques élémentaires, appelées segments. Ces segments sont eux-mêmes constitués de composants, tels que des trous à sauter, des ennemis à vaincre, ou encore des pièces à collecter par exemple. Pour générer des morceaux de niveau adaptés au niveau du joueur, Jennings-Teats et al. ont besoin de deux éléments : un modèle de difficulté qui indique ce qui rend un niveau facile ou difficile, et un modèle dynamique de la performance du joueur qui montre s'il a besoin d'aide ou de challenge supplémentaires.

Pour le modèle de difficulté, Jennings-Teats et al. ont créé un outil de collecte de données du comportement du joueur lors d'une phase de jeu sur un court segment d'environ dix secondes. Ces données sont alors utilisées comme attributs lors du machine learning (ML). Après le jeu, le niveau est noté de 1 à 6 par le joueur. Les segments de niveau sont générés par un algorithme basé sur le rythme désiré des actions (telles que courir, sauter, ou attendre), sans restriction sur le rythme.

Pour l'apprentissage, plusieurs attributs sont considérés tels que le nombre d'apparition d'un composant de niveau, le nombre d'apparition de deux composants adjacents dans un même segment, le nombre de trous, la largeur moyenne d'un trou, le changement global d'altitude des plateformes, les largeurs des plus petites et plus grandes plateformes. Cette liste n'est pas exhaustive, les possibilités de personnalisation de la génération étant presque infinies.

Pour le modèle dynamique de la performance du joueur, certains attributs sont également amassés par l'outil de collecte, tels que le temps passé à reculer ou attendre, le temps total de complétion du segment, le nombre de pièces collectées, et l'issue de la tentative : si le joueur est mort ou a complété le segment.

En se basant sur les attributs de description des niveaux, une classification SVM (Support Vector Machine, une technique de classification supervisée) est appliquée pour obtenir une liste classifiée des segments générés pendant le jeu. Dans le même temps, les attributs de comportement du joueur sont collectés et évalués sur un modèle entraîné sur les données de l'outil de collecte. Ainsi, avant que le joueur ne progresse au segment suivant, un segment est sélectionné depuis la liste en fonction de la performance courante du joueur, ce qui permet de garder un challenge constant, puisque les segments sont toujours sélectionnés en fonction des performances courantes, donc mises à jour, du joueur. Ce comportement s'applique aussi si le joueur perd, et dans ce cas la difficulté est réduite par la sélection de segments plus faciles.

L'autre approche d'un système de difficulté adaptative par apprentissage se base sur des profils uniques d'utilisateur, tel que l'on peut en rencontrer en rééducation post-AVC. C'est le cas de la technique proposée par Hocine et al. [10].

Le jeu thérapeutique consiste en une tâche de pointage, réalisée avec la souris d'une tablette graphique. Les utilisateurs effectuent des mouvements du bras sur le plan où est installé le dispositif (Fig. 2.4).

Un niveau est constitué de plusieurs scènes. L'objectif de l'utilisateur est d'atteindre la fin du niveau, et ramasser la gemme qui symbolise le succès de la tâche. Pendant une session, l'utilisateur parcourt les scènes en triomphant des adversaires qui bloquent sa progression. Il peut également ramasser des pièces pour augmenter son score.

Les niveaux sont générés pendant la session. Un niveau est composé d'un ensemble de scènes, qui se débloquent au gré de la progression de l'utilisateur. Les adversaires sont vaincus au moyen d'outils piochés dans son inventaire. Pour les encourager à effectuer des mouvements de plus grande amplitude, les pièces d'or, situées plus loin, rapportent plus de points que les pièces d'argent, qui elles-mêmes en rapportent plus que celles de bronze (Fig. 2.5).



FIGURE 2.4 – Dispositif de rééducation de Hocine et al. [10]



FIGURE 2.5 – Exemple de scène.

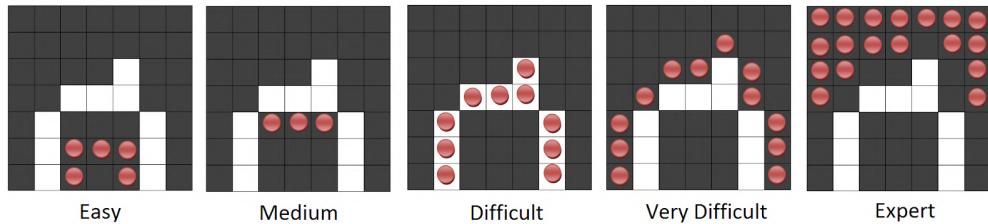


FIGURE 2.6 – Les cinq niveaux de difficulté.

Le processus d'adaptation se découpe en quatre phases. Tout d'abord, une calibration est effectuée afin de construire le profil de l'utilisateur. Ensuite, le module d'entraînement génère des séquences de tâches de pointage adaptées à ce profil. Le module de génération procédurale utilise alors ces séquences pour générer un niveau de jeu. Lorsque le niveau est parcouru, la difficulté est adaptée dynamiquement aux performances de l'utilisateur. Ces dernières sont enfin intégrées pour mettre à jour son profil.

Pour commencer, le profil de l'utilisateur est basé sur la prédiction de ses capacités à court terme et de sa condition physique quotidienne. Pour cela, il est nécessaire de connaître son amplitude de mouvement, dans sa “zone de capacité”. Cette zone est modélisée par une matrice, qui sert d'interface entre la zone réelle de travail et l'environnement virtuel. Chaque cellule de la matrice contient des informations sur les mouvements réalisés par l'utilisateur. En bougeant la souris dans la zone de travail, l'utilisateur permet au système d'évaluer sa zone de capacité, par exemple en récoltant les coordonnées. La zone de capacité est définie pendant la calibration, où la zone initiale de l'utilisateur est créée, puis mise à jour pendant la session.

Ensuite, le module d'entraînement génère une liste de tâches de pointage et la fournit au générateur de niveaux. La génération de tâches est faite avant la session de jeu. Les scènes sont générées pour chaque mode de difficulté, et la transition entre ces modes est faite selon les performances de l'utilisateur, de manière dynamique pendant la session.

La première étape consiste à générer les tâches de pointage pour chaque niveau de difficulté. La difficulté est définie par la direction du mouvement et la distance entre le point de départ et la cible à atteindre. La matrice précédemment établie est traitée : la bordure de la zone de capacité du patient est récupérée, et représente le niveau de difficulté “difficile” (parmi Facile, Moyen, Difficile, Très difficile et Expert). La génération se fait donc en variant les paramètres définissant la difficulté, selon le niveau de difficulté : les niveaux en-dessous de difficile voient les cibles plus proches de la bordure, ceux au-dessus voient leurs cibles plus éloignées de la bordure (Fig. 2.6).

La seconde étape vise à sélectionner les tâches de pointage pour un niveau de jeu. Le nombre de tâches générées étant très important, il faut un processus de sélection de

quelques tâches, pour les incorporer dans le niveau. La contrainte est la suivante : il faut qu'au fil des tâches, l'utilisateur gagne en mobilité. L'algorithme la réalisant est l'arbre de recherche de Monte Carlo (MCTS), qui consiste à chercher des solutions optimales dans un arbre construit à partir d'une sélection aléatoire dans l'espace des solutions. Un arbre est alors généré pour chaque niveau de difficulté.

Le module d'entraînement propose alors des tâches de pointage en accord avec les performances de l'utilisateur. Ses taux de succès et d'échecs sont notamment utilisés dans la décision d'ajuster le niveau de difficulté. Le thérapeute possède également des moyens de contrôler la session et donc l'adaptation : il peut sélectionner le jeu, le nombre de cibles pour chaque niveau et mettre à jour la zone de capacité de l'utilisateur, en termes de dimension et de précision de mouvement.

La troisième phase est la génération des niveaux de jeu. Le contenu est généré de manière procédurale, adaptée et cohérente. Les séquences de tâches générées par le module d'entraînement sont utilisées comme entrées pour la génération de scènes, pour déterminer la position des cibles à atteindre. Ensuite, le système complète les scènes, en ajoutant des connecteurs entre les éléments, qui favorisent la progression de l'utilisateur.

Enfin, les performances de l'utilisateur sont en permanence mises à jour dans son profil, afin de générer des tâches, puis des scènes et enfin des niveaux qui soient en adéquation avec son expertise du jeu thérapeutique.

Les méthodes par apprentissage requièrent donc un grand volume de données pour bâtir des modèles d'entraînement robustes et viables. Ces données peuvent être extraites d'une base d'utilisateur générale, sans personnalisation dans la collecte d'information. Elles peuvent à l'inverse résulter d'une calibration individuelle et focalisée sur un individu en particulier, apportant ainsi une notion de personnalisation au modèle.

2.4 Méthodes prédictives et probabilistes

Les méthodes prédictives et probabilistes sont deux types de méthodes différentes, qui possèdent des points communs dans leur approche du calcul. Les méthodes prédictives annoncent en avance la performance de l'utilisateur, en se basant par exemple sur ses performances passées tels que le présentent Nirme et al. [5].

Les méthodes probabilistes quant à elles quantifient les chances de chaque issue de se produire. Plusieurs procédés existent, parmi lesquels le système Hamlet, conçu par Hunicke et Chapman [1], qui observe plusieurs variables du jeu pour estimer les chances de réussite ou d'échec de l'utilisateur.

Pirovano et al. [6] comparent une méthode par seuil classique avec une méthode probabiliste basée sur la Quête Bayésienne, qui vise à inférer la probabilité d'un événement en

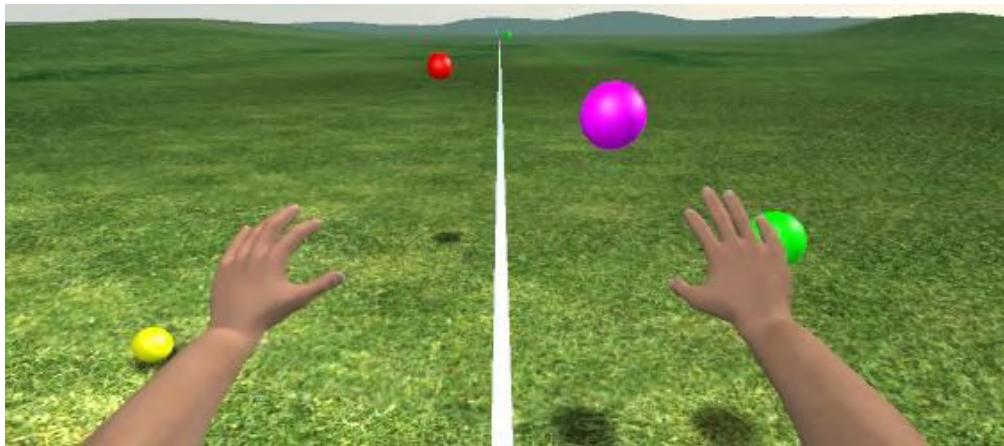


FIGURE 2.7 – Exemple d'un niveau du RGS

fonction de celles d'événements passés.

Nirme et al. [5] proposent de comparer deux approches : l'une basée sur une recherche aléatoire, l'autre sur une recherche prédictive basée sur un profil utilisateur. Leur système, appelé le RGS pour Rehabilitation Gaming System, ou Système de Rééducation par le jeu en français, est constitué d'une part d'un outil de suivi du bras, du coude et des doigts, à l'aide d'une caméra, de patchs colorés et de gants de données respectivement. D'autre part, il propose des tâches individualisées, et variables. A l'écran, les utilisateurs voient des sphères de couleur approcher à différents vitesses, intervalles et décalages vers la gauche ou la droite, par rapport au centre. Leur tâche est de toucher les sphères, avec leur main gauche ou droite respectivement (Fig. 2.7).

La difficulté est régulée au moyen de plusieurs paramètres : vitesse des sphères, écart par rapport au centre, taille des sphères et intervalle temporel entre deux apparitions. La performance du joueur, définie comme le pourcentage de sphères touchées sur chaque côté, contrôle la difficulté. L'objectif est de proposer des paramètres pour lesquels la performance du joueur sera proche de la performance cible. Pour ce faire, deux approches sont explorées : une recherche aléatoire et une recherche prédictive.

Pour la recherche aléatoire, le problème est défini comme une recherche locale dans un espace de recherche défini par les limites de définition des paramètres de difficulté. Il est résolu par un algorithme de marche aléatoire, adapté pour éviter les extrema locaux. Il se déplace en escalier dans l'espace de recherche selon une direction aléatoire jusqu'à ce que l'erreur de la performance résultante par rapport à la performance cible dépasse un certain seuil. La hauteur d'une marche est modulée par l'amplitude de l'erreur de performance.

Pour la recherche prédictive, une fonction de prédiction des performances de l'utilisa-

teur est employée. Elle s'appuie notamment sur un profil utilisateur défini au préalable. Ce profil est ensuite utilisé pour générer aléatoirement de nouveaux paramètres dans le sous-espace où le modèle prédit une performance proche de la performance cible. Lorsque la performance prédite est trop éloignée de la performance cible, les nouveaux paramètres sont choisis d'une manière similaire à celle de l'algorithme de recherche aléatoire.

Les paramètres de difficulté sont indépendants du côté travaillé dans le jeu thérapeutique, ce qui permet aux algorithmes de gérer la gauche et la droite séparément, améliorant ainsi l'individualisation de la tâche.

Parmi les méthodes probabilistes, le système Hamlet, conçu par Hunicke et Chapman [1], est considéré comme le premier système de difficulté adaptative dynamique, i.e. dont l'adaptation est faite en même temps que l'utilisateur effectue ses tâches. Hamlet a été conçu au départ pour le genre FPS (First Person Shooter, jeu de tir en vue subjective), et il est implémenté dans le moteur de jeu du jeu vidéo Half-Life, développé et édité par Valve Corporation.

En se basant sur de nombreux paramètres observés pendant la session de jeu, tels que la position de l'utilisateur dans le niveau, son avancement global, ses points de vie ou encore ses munitions, Hamlet essaie d'inférer la probabilité d'échec de l'utilisateur lors de la prochaine action, et ainsi entreprendre des actions préventives pour que la situation ne se produise pas. L'objectif derrière cette analyse est de proposer constamment une épreuve ni trop difficile, i.e. qui demande un niveau de compétence trop élevé, ni trop faible, conduisant à un ennui de l'utilisateur (Fig. 2.8).

L'atteinte de cet objectif se déroule en trois étapes : évaluer quand un ajustement est nécessaire, déterminer quels changements doivent être faits et les exécuter de manière aussi fluide que possible.

L'évaluation est bâtie sur la variation de l'inventaire de l'utilisateur. L'inventaire est défini comme les ressources à la disposition de l'utilisateur, telles que la santé ou les munitions. Les variations d'inventaire sont observées par Hamlet, qui déduit des distributions de probabilité des différents résultats. Deux types d'ajustement sont alors envisagés, aux effets distincts voire contraires.

Les actions réactives vont ajuster des éléments déjà présents dans le jeu, tels que par exemple la précision des ennemis, ou leur force. Leur exécution est immédiate et directement liée à l'expérience de l'utilisateur au moment où elles sont exécutées, ce qui peut influencer la perception et l'immersion de l'utilisateur. A l'inverse, les actions proactive impactent des éléments qui ne sont pas encore instanciés, tels que l'ordre d'apparition des ennemis, ou leur fréquence. L'impact de ces changements est plus éloigné dans le temps, ce qui peut mettre en doute leur efficacité, et sont donc plus difficiles à estimer.

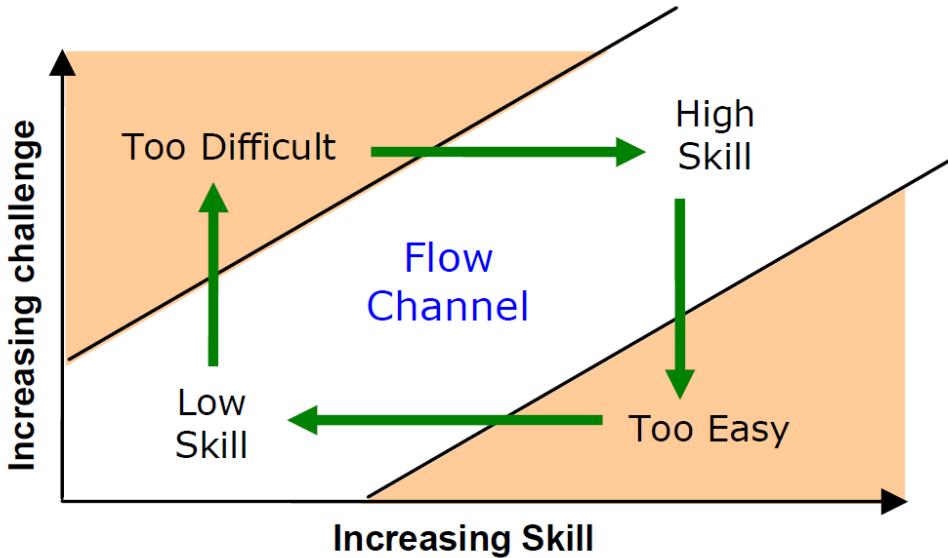


FIGURE 2.8 – Objectif du système Hamlet.

Au final, pour compenser les effets négatifs de chaque type, un compromis doit être fait, qui consiste en deux stratégies d'ajustement. La première, la stratégie de la zone de confort, garde les utilisateurs actifs et alertes, sans toutefois les forcer au challenge. Hamlet interviendra fréquemment, quoique laissant la place à des erreurs. La seconde, la stratégie de la zone de malaise, est conçue pour des utilisateurs plus expérimentés aimant être en permanence sur le point d'échouer. Les aides fournies par Hamlet seront rares et précieuses.

Le système présenté par Hunicke et Chapman a prouvé être fiable, souple et riche, et est aujourd'hui considéré comme fondateur dans le domaine de la difficulté adaptative dynamique. Il a inspiré de nombreux systèmes, de par son approche probabiliste, ou par son traitement individualisé des ressources allouées au joueur.

Pirovano et al. [6] présentent d'ailleurs une étude comparative de deux méthodes : la première fait appel à un simple seuil de performance, et ne sera donc pas abordée dans la suite de cette partie, tandis que l'autre est fondée sur une méthode adaptative dérivée de l'inférence bayésienne.

Le système IGER (pour Intelligent Game Engine for Rehabilitation, moteur de jeu intelligent pour la réhabilitation en français) est composé de deux parties : le moteur de jeu et le contrôleur. Le rôle du contrôleur est de gérer toute la partie logique et calcul d'IGER : évaluer et adapter les paramètres, au moyen de la base de connaissances, fournir des informations au thérapeute, retourner des données au moteur.

Le moteur de jeu est le moyen employé par l'utilisateur pour accomplir la tâche demandée (Fig. 2.9).

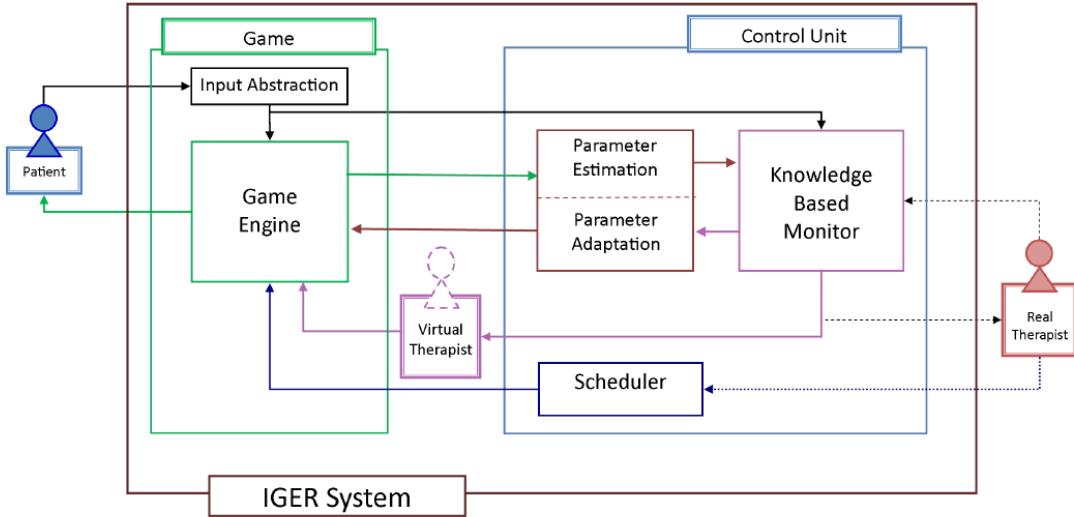


FIGURE 2.9 – Composition du système IGER

La méthode de l’inférence bayésienne consiste à adapter un seuil de performance en fonction des essais passés. Les essais ont systématiquement une issue binaire : succès ou échec.

Pirovano et al. ont modifié cette méthode pour qu’elle puisse se réaliser en temps réel, i.e. faire une adaptation après chaque essai.

Avant le début d’une session, deux nombres sont définis : $x = [x_1, x_2, \dots, x_n]$ le vecteur rassemblant les n paramètres d’adaptation, et p_{end} la performance cible. x_{end} est alors le vecteur cible, qui représente les paramètres adaptés pour lesquels la performance cible est atteinte. Ce nombre est l’objectif de cette méthode. A chaque essai, la probabilité d’obtenir un succès ou un échec est quantifiée, en se basant sur l’historique des issues des précédents essais. Après chaque essai, une modification des paramètres est effectuée pour que la performance cible p_{end} ait une probabilité maximale d’être atteinte lors du prochain essai.

Les méthodes prédictives et probabilistes utilisent donc l’incertitude du résultat comme moteur de leurs approches. Elles prennent appui sur l’étude des performances passées pour certaines, ou directement de la performance courante de l’utilisateur pour d’autres. Ces méthodes demandent des calculs plus poussés que les méthodes par seuils, sans toutefois exiger des volumes de données importants tels que ceux des méthodes par apprentissage.

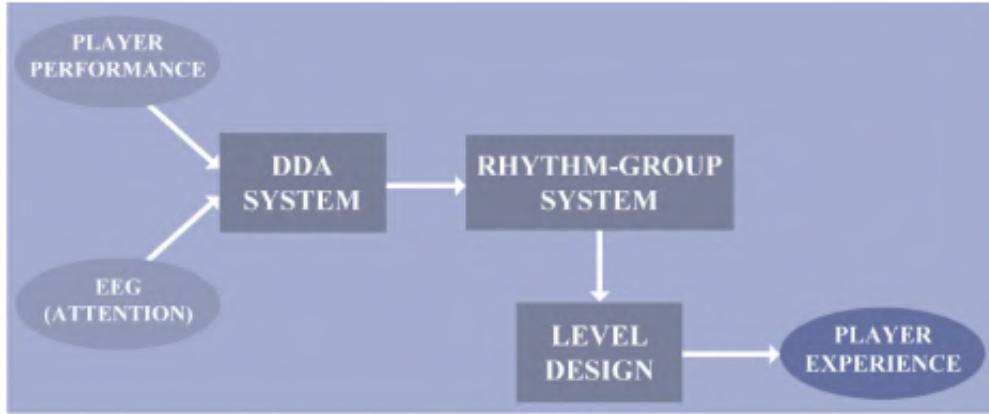


FIGURE 2.10 – Système combiné d’adaptation de la difficulté de Mikami et al. [12]

2.5 Méthodes par scores

Les méthodes par scores consistent à définir puis calculer des scores, de différents éléments, comme par exemple des scores de performance [12], ou d’entités liées à la scène, comme le nombre d’adversaires restants ou la vie du joueur [11]. L’adaptation prend alors place en tenant compte des scores de ces éléments, de manière individuelle ou collective.

Mikami et al. [12] présentent une méthode qui propose de combiner trois composants : un système de difficulté adaptative dynamique (DDA), la théorie des groupes de rythmes et une interface neuronale directe (IND ou BCI, Brain Computer Interface). La théorie des groupes de rythmes (Rhythm-group Theory en anglais) est une technique de génération automatique de niveaux, qui crée une sensation de rythme lors de la phase de jeu. L’IND utilisée capture les données d’EEG (électroencéphalographie) du joueur pour les amener directement dans le système DDA (Fig. 2.10).

Ces données portent sur l’attention que le joueur dédie à la tâche en cours, à raison d’une donnée par seconde. Le calcul est fait en interne par le BCI, qui renvoie une valeur entre 0 (inattentif) et 100 (concentré). Mikami et al. calculent alors la moyenne des mesures réalisées dans le niveau afin de n’avoir plus qu’une seule donnée à envoyer au système de difficulté adaptative. L’autre entrée consiste en les données de performance de l’utilisateur, calculée selon deux paramètres : le nombre de morts ou de coups reçus, et le temps de jeu. Chaque paramètre se voit affecté d’un poids, qui est ajusté à plusieurs reprises ensuite afin d’en étudier les effets.

Les données de performance et d’attention sont ensuite combinées pour obtenir une valeur globale pour décider de l’ajustement de la difficulté. Le système de groupes de rythmes reconnaît plusieurs types d’éléments, tels que “saut court”, “courir”, “saut long”,

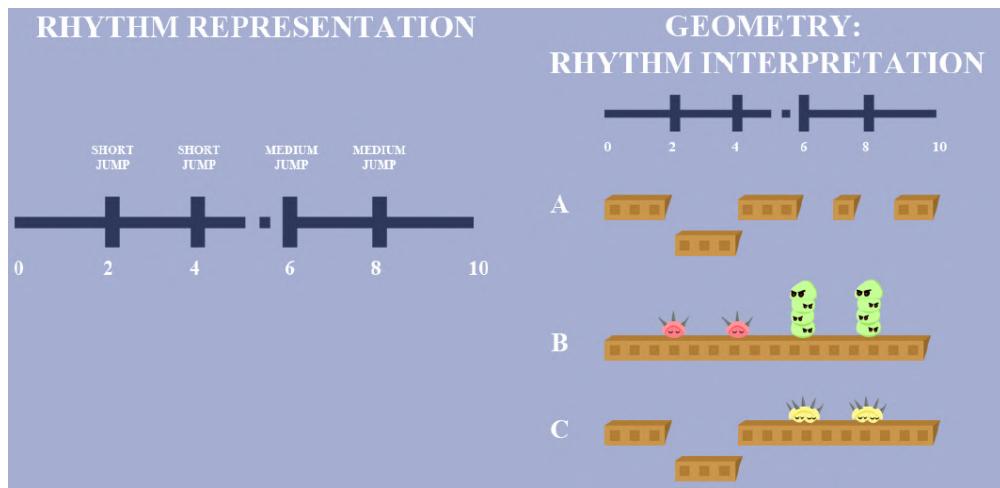


FIGURE 2.11 – Représentation d'un rythme (gauche) et différentes interprétations géométriques (droite)

etc. Les scores de performance et d'attention sont calculés pour chaque type d'élément, afin de savoir combien de ceux-ci seront générés dans le niveau suivant. Le système générera plus d'éléments lorsque l'utilisateur a tendance à réussir, afin de privilégier l'amélioration continue de sa performance. À l'inverse, il diminuera la fréquence d'apparition des éléments que l'utilisateur rate. Ces valeurs évoluent avec la performance globale de l'utilisateur tout au long de la session.

Toutes ces valeurs sont ensuite injectées dans le système de génération rythmée, qui va alors finaliser le niveau à générer (Fig. 2.11). Ce système considère quatre paramètres importants :

- type d'action : courir et sauter sont les plus simples.
- type de rythme : le seul type utilisé ici est le rythme régulier, c'est-à-dire qu'il distribue également tous les types d'action en son sein.
- densité du rythme : nombre d'actions dans un rythme, entre 5 et 50 actions. Le nombre d'actions choisi dépend des performances de l'utilisateur : plus il est performant, plus la densité d'actions du rythme est importante.
- longueur du rythme : temps de jeu prévu du niveau, entre 5 et 30 secondes. Il dépend également des résultats de l'utilisateur, selon le même principe que la densité du rythme.

La difficulté du niveau suivant est calculée d'après la difficulté précédente ajoutée d'une variation du score global de l'utilisateur. Cette variation est positive ou négative, en fonction de l'augmentation ou de la diminution de la difficulté.

Enfin, un niveau totalement jouable est obtenu en sortie du système d'interprétation géométrique. Ce niveau tient donc compte de la difficulté calculée, du nombre d'occurrences de chaque type d'élément, du rythme qui a été défini pour ce nouveau niveau afin de répondre aux performances de l'utilisateur.

Sutoyo et al. [11] détaillent quant à eux un système complexe, basé d'une part sur l'évaluation de la performance du joueur, au travers de trois conditions ; d'autre part en appliquant des coefficients multiplicateurs à plusieurs entités et variables du jeu afin d'en modifier le comportement.

L'évaluation de la performance du joueur se calcule par les trois conditions suivantes :

- le nombre de vies restantes du joueur, i.e. du château ou de l'entité à protéger (cas des jeux de type Tower Defense). Si les vies diminuent trop, le système réduit la difficulté, et vice versa.
- le niveau de santé des ennemis. Lorsqu'un ennemi de type boss de niveau est tué, le joueur gagne 5 vies supplémentaires. Si la santé cumulée des ennemis restants est supérieure à 50% de la santé totale, le système considère que le joueur échoue et diminuera la difficulté lors de la prochaine vague.
- les capacités passives (points de compétence). Le joueur a accès à des capacités passives de trois types, grâce à l'attribution de points de compétences en attaque, défense et support.

Les coefficients multiplicateurs sont au nombre de quatre : statut, apparition, or et points de difficulté globale. Ces coefficients ont une influence directe sur les variables, au nombre de trois : points de statut, affectant la puissance des ennemis, points d'apparition, régissant le nombre d'ennemis du niveau en cours, et points d'or, incident sur la quantité d'or reçue après avoir terminé un niveau. Le coefficient "points de difficulté globale" affecte toutes les variables à la fois.

Pour chaque condition d'évaluation de la performance du joueur, les coefficients et les variables sont mis à jour après chaque vague, par leur augmentation ou leur diminution. Par exemple, si le joueur perd entre 4 et 7 vies, les coefficients de statut et d'apparition seront réduits, celui d'or sera augmenté et les points de statut et d'apparition seront diminués. Des règles similaires sont appliquées pour le niveau restant de santé des ennemis et les capacités passives utilisées par le joueur.

Ainsi, pendant chaque vague puis à la fin de chacune, les trois conditions sont évaluées, afin de connaître les nouveaux coefficients et variables de la vague suivante. Les performances du joueur impactent directement la difficulté, puisque les coefficients et variables peuvent recevoir des valeurs positives et/ou négatives.

Les entités affectées par ces coefficients et variables sont les suivantes : le nombre d'ennemis de la prochaine vague (minimum 20), le bonus d'or obtenu en fin de vague (différent

	Contexte	Méthode	Paramètres observés	Paramètres modifiés
Hunicke et Chapman	JV en général	Probabiliste	position, progression, fréquence de mort, nombre d'essais du segment, interventions précédentes du système	précision et force des attaques, force des armes, niveau de santé, type d'ennemis, ordre d'apparition, santé, dommages, ...
Ma et Bechkoum	Rééducation post-AVC membre supérieur	Seuils	précision, temps de complétion	nombre d'éléments à l'écran, temps imparti, vitesse des cibles
Alankus et al.	Rééducation post-AVC membre supérieur	Seuils	"performance du joueur" (pas de détails)	pas de détails
Jennings-Teats et al.	JV de type plateformes	Apprentissage	temps de complétion, temps d'attente, nombre de pièces, succès ou échec	segments de niveau
Nirme et al.	Rééducation post-AVC membre supérieur	Préditive	pourcentage de cibles atteintes	vitesse, portée, taille et intervalle d'apparition des objets
Pirovano et al.	Rééducation en général	Probabiliste	pourcentage de cibles atteintes	taille du panier, taille et vitesse des fruits, portée horizontale, nombre de fruits
Hristova	JV de type FPS	Seuils	activité cardiaque et électrodermique, température, excitation, anxiété, progrès, performances	coût des actions, intensité des stimuli, ressources...
Li et al.	JV de type plateformes	Seuils	nombre d'obstacles passés, nombre d'échecs dans le même niveau	apparition d'obstacles, de bonus
Hocine et al.	Rééducation post-AVC membre supérieur	Apprentissage	amplitude maximale de mouvement, nombre de succès et d'échecs de pointage	niveau de difficulté
Sutoyo et al.	JV de type tower defense	Scores	vies du joueur, santé totale des ennemis, points de compétence	force, armure et nombre d'ennemis, or reçu en fin de vague
Mikami et al.	JV en général	Scores	EEG (attention), nombre de morts et temps de jeu	type d'action, nombre d'actions de chaque type, longueur d'un niveau

TABLE 2.1 – Bilan – Méthodes d'adaptation de la difficulté

de l'or obtenu de toute manière), la vie maximale des ennemis et la résistance de l'armure des ennemis.

En plus de ce système complexe, des règles supplémentaires ont été ajoutées par Sutoyo et al. pour équilibrer le jeu, comme par exemple l'attribution additionnelle de points de statut et d'apparition au début du jeu, pour "entraîner" le joueur.

En conclusion, les méthodes par scores permettent d'établir des systèmes fins et complexes, capables d'affiner la difficulté à des niveaux très bas. Ils nécessitent cependant de nombreux éléments, tels que des variables et des coefficients, qui impactent directement le gameplay et les performances de l'utilisateur. Ces éléments sont indispensables pour mettre en place une telle méthode, et demandent donc que l'application soit déjà suffisamment complexe.

2.6 Bilan

De cette étude, nous pouvons extraire deux réflexions, résumées en deux tableaux. Tout d'abord, le tableau 2.1 montre qu'il existe de nombreuses manières de procéder à l'adaptation du niveau de difficulté d'un jeu, ou d'une tâche. Parmi les articles mentionnés, cinq ont pour contexte la rééducation motrice post-AVC du membre supérieur. Cependant, les méthodes mises en place ne conviennent pas à notre situation : certaines

	Type	Calibration	Contexte	Méthode
Hunicke et Chapman	Online	Non	JV en général	Probabiliste
Ma et Bechkoum	Online	Oui	Rééducation post-AVC membre supérieur	Seuils
Alankus et al.	Online	Oui	Rééducation post-AVC membre supérieur	Seuils
Jennings-Teats et al.	Online	Oui	JV de type plateformes	Apprentissage
Nirme et al.	Online	Oui	Rééducation post-AVC membre supérieur	Préditive
Pirovano et al.	Online	Oui	Rééducation en général	Probabiliste
Hristova	Online	Non	JV de type FPS	Seuils
Li et al.	Online	Non	JV de type plateformes	Seuils
Hocine et al.	Online	Oui	Rééducation post-AVC membre supérieur	Apprentissage
Sutoyo et al.	Online	Non	JV de type tower defense	Scores
Mikami et al.	Online	Oui	JV en général	Scores

TABLE 2.2 – Bilan – Type d'adaptation et calibration

sont trop simples, telle que les méthodes par seuils employées par Ma et Bechkoum [2], Alankus et al.[3] par exemple. Notre jeu thérapeutique faisant travailler plusieurs caractéristiques du patient, il est nécessaire de pouvoir ajuster tous les aspects travaillés. D'autres méthodes sont trop complexes, telles que celles par scores de Sutoyo et al. [11] et Mikami et al. [12]. Les données et formules utilisées suggèrent une richesse d'entités et d'objets dans l'environnement virtuel que nous n'avons pas. Enfin, les méthodes par apprentissage telle que celle de Jennings-Teats et al. [4] demandent beaucoup plus d'informations et de données que nous pourrons en générer, notamment par le fait que la population de notre projet est très hétérogène : les patients sont âgés de 18 à 80 ans, chacun ayant des incapacités qui lui sont propres et qui au demeurant peuvent varier drastiquement d'un jour à l'autre.

Ensuite, le tableau 2.2 présente les différentes méthodes en s'intéressant plus particulièrement à la présence d'une calibration ainsi qu'au type d'adaptation utilisé. Une adaptation de type "online" se déroule pendant la phase de jeu ou de rééducation, alors que l'utilisateur se sert du système. A l'inverse, une adaptation de type "offline" s'effectue entre les phases de jeu ou de rééducation, lorsque l'utilisateur n'est pas en train de remplir ses objectifs ou ses tâches. On remarque immédiatement que l'ensemble des so-

lutions retenues sont de type "online".

Quant à la calibration, elle peut être présente ou ne pas l'être, selon la manière dont les chercheurs ont conçu leur système. Nous notons cependant que lorsqu'il s'agit d'un contexte de rééducation, qu'elle soit générale ou ciblée, la phase de calibration est systématiquement présente. Les équipes n'ayant pas mis en place une phase de calibration se trouvent par ailleurs dans des contextes plus généraux d'application, pour lesquels il est moins important d'être précis dès le lancement du système.

Il est donc essentiel de créer un modèle approprié, prenant appui sur cette étude tout en mettant le patient et sa rééducation en son centre : l'objectif est de créer un modèle d'adaptation de la difficulté prenant en compte les capacités du patient, chaque jour, et de proposer des tâches adéquates en tous temps, toutes situations.

Chapitre 3

Modèle de description et d'adaptation de la difficulté

3.1 Introduction

Nous souhaitons aboutir à un modèle d'adaptation de la difficulté qui prenne en compte tous les aspects de chaque patient, notamment l'ensemble des caractéristiques qu'il travaille, et également son état de santé, qui peut varier d'un jour à l'autre. Ce modèle doit pouvoir changer dynamiquement la difficulté des tâches proposées au patient, de manière transparente et homogène pour celui-ci (le type d'adaptation est donc online).

Ce chapitre décrit de manière exhaustive et détaillée le modèle de description et d'adaptation de la difficulté qui a été conçu, développé et implémenté dans le cadre du jeu de l'oiseau, qui fait partie du projet plus global "*Cesaar-AVC*" décrit en section 1.1.2.

La figure 3.1 résume ce modèle. Il est constitué de trois principaux composants.

Le modèle de description de la difficulté, ou modèle de difficulté pour la rééducation, est le premier composant. Il se décompose lui-même en plusieurs éléments, tels que les définitions, présentées en section 3.2, qui permettent d'établir les aptitudes évaluées du patient pendant le jeu thérapeutique. La section 3.3 explique ensuite comment la description est réalisée et comment la difficulté d'un exercice est quantifiée.

La calibration du système est le second composant, présentée en section 3.3.1. Elle s'effectue à chaque début de séance afin d'initialiser le système avec les performances et les capacités du jour du patient.

La session est le troisième composant. La session représente la phase active de la rééducation, pendant laquelle le patient effectue les tâches demandées. Elle est elle-même constituée de quatre phases qui se succèdent.

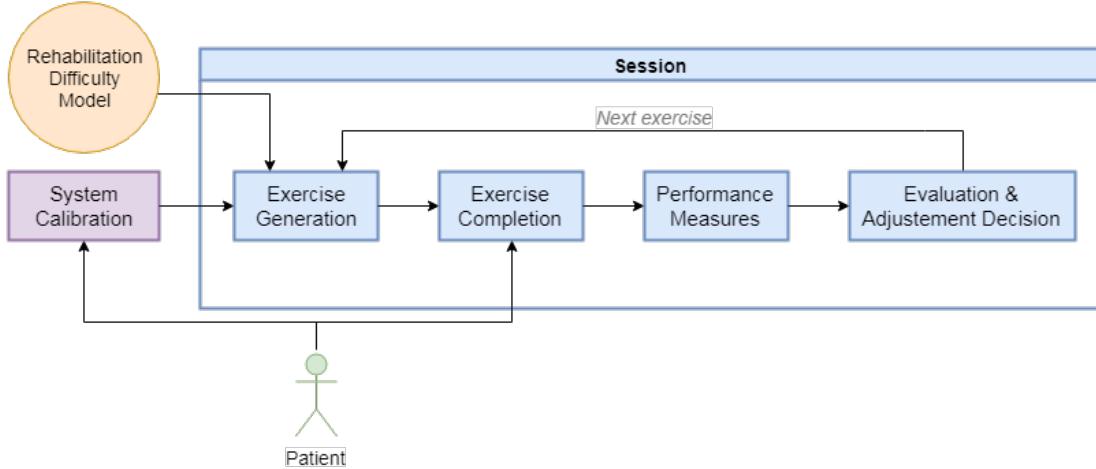


FIGURE 3.1 – Modèle d’adaptation de la difficulté

Tout d’abord, le système procède à une génération de l’exercice à venir (section 3.4). La constitution de cet exercice va dépendre de plusieurs paramètres, telles que les contraintes appliquées par le modèle de difficulté et aussi par les résultats précédents du patient. Deux approches furent suivies pour permettre la génération d’exercices : la première, décrite en section 3.4.2, permet de générer un exercice autour d’un niveau donné en argument de difficulté. C’est cette solution qui fut implémentée dans le jeu de l’oiseau. La deuxième approche est apparue plus tôt dans le projet, mais n’a pas complètement abouti (section 3.4.3) et est toujours en réflexion.

Ensuite, la deuxième phase est celle où le patient va compléter l’exercice, en répondant aux tâches demandées. A l’heure actuelle, ces tâches se résument à ramasser toutes les pièces qui lui sont présentées. Ses performances sont mesurées en temps réel, pendant la complétion de l’exercice. Enfin, le système évalue la performance du patient par rapport à la consigne (section 3.5.1) puis décide de l’adaptation à réaliser (section 3.5.2), afin d’apporter les informations les plus pertinentes possibles pour la génération de l’exercice suivant.

Les phases se répètent alors, à partir de la génération d’un nouvel exercice.

3.2 Exercices, aptitudes et autres définitions

Cette section définit les notions abordées tout au long de ce chapitre.

3.2.1 Exercice

Un exercice est une suite de 20 pièces, sur 20 lignes et 5 colonnes, où la première et la vingtième pièces sont imposées en colonne 1 (départ et arrivée en zone de repos). Le patient réalise quotidiennement une série de plusieurs exercices. La consigne est de ramasser toutes les pièces.

3.2.2 Aptitudes

Deux définitions sont proposées pour chacune : l'une générale concernant le mouvement de pronation-supination, l'autre appliquée au jeu de l'oiseau. Les exemples sont illustrés sur la figure 3.2.

Amplitude [A]

Définition générale : consiste à atteindre et tenir les amplitudes maximales et approchantes.

Définition appliquée : aptitude à aller ou rester dans les colonnes 4 et 5. Deux niveaux d'amplitude sont possibles : *A4* (colonne 4) et *A5* (colonne 5).

Maintien [M]

Définition générale : consiste à maintenir une position, à un angle fixe et pendant un temps donné.

Définition appliquée : aptitude à enchaîner des pièces placées dans la même colonne, à l'exception de la colonne 1 qui correspond à la zone de repos. Un maintien en colonne 1 ne correspond donc pas à un mouvement.

Vitesse [V]

Définition générale : consiste à parcourir une amplitude (distance ou angle) importante en un temps réduit.

Définition appliquée : aptitude à atteindre la cible suivante (sur la ligne suivante) éloignée d'au moins deux colonnes de la position actuelle. Trois niveaux de vitesse sont possibles : *V2* (saut de 2 colonnes), *V3* (saut de 3 colonnes) et *V4* (saut de 4 colonnes).

Précision [P]

Définition générale : consiste à atteindre une cible dans une position non extrême, différente de la position précédente.

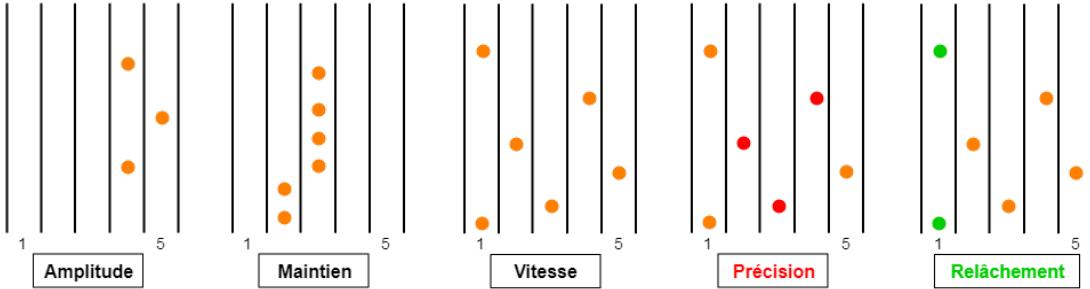


FIGURE 3.2 – Exemples appliqués des aptitudes.

Définition appliquée : aptitude à atteindre une cible dans une colonne intermédiaire différente de la colonne précédente.

Relâchement [R]

Le relâchement se caractérise par un déplacement en colonne 1. Le relâchement n'est pas une aptitude directement travaillée par le patient, car il lui suffit de relâcher la tension musculaire pour atteindre cette position. Cependant c'est une caractéristique à prendre en compte lors de l'évaluation de la difficulté de l'exercice.

Huit niveaux d'aptitudes sont travaillés au total :

$$Apts = \{A4, A5, M, V2, V3, V4, P, R\}$$

3.2.3 Transitions

Transition

Une transition est un couple de colonnes qui décrit le passage d'une pièce sur la ligne k à une pièce sur la ligne $k + 1$. Puisque l'outil est composé de 5 colonnes, il existe 25 transitions possibles :

$$t_{i \rightarrow j} = \{l_k \rightarrow l_{k+1}, k \in [1; 19] \wedge col_i \rightarrow col_j, i, j \in [1; 5]\} \quad (3.1)$$

Transitions

Les transitions considérées pour les exercices sont l'ensemble défini ci-dessus, privé de la transition $t_{1 \rightarrow 1}$. Cette transition correspond à un maintien en colonne 1, zone de repos, qui n'est pas pris en compte. Il reste 24 transitions possibles pour les exercices :

$$T = \{(i, j) \in [1; 5]^2 - (1, 1) | t_{i \rightarrow j}\} \quad (3.2)$$

Un exercice est donc décrit par une suite de transitions.

Transitions	Configurations	Transitions	Configurations
$t_{1 \rightarrow 2}, t_{2 \rightarrow 3}, t_{3 \rightarrow 2},$ $t_{4 \rightarrow 3}, t_{5 \rightarrow 4}$	P	$t_{3 \rightarrow 1}$	$V2$
$t_{1 \rightarrow 3}, t_{4 \rightarrow 2}, t_{5 \rightarrow 3}$	$V2P$	$t_{4 \rightarrow 1}$	$V3$
$t_{5 \rightarrow 2}$	$V3P$	$t_{5 \rightarrow 1}$	$V4$
$t_{3 \rightarrow 4}$	$A1P$	$t_{2 \rightarrow 4}$	$V2A4P$
$t_{2 \rightarrow 2}, t_{3 \rightarrow 3}$	M	$t_{1 \rightarrow 4}$	$V3A4P$
$t_{4 \rightarrow 4}$	$A4M$	$t_{3 \rightarrow 5}$	$V2A5$
$t_{5 \rightarrow 5}$	$A5M$	$t_{2 \rightarrow 5}$	$V3A5$
$t_{4 \rightarrow 5}$	$A5$	$t_{1 \rightarrow 5}$	$V4A5$

TABLE 3.1 – Correspondance des transitions et configurations.

3.2.4 Configurations

Une configuration correspond à des aptitudes seules ou combinées, réalisées lors des transitions. Il existe 16 configurations possibles. Une configuration correspond à une ou plusieurs transitions. Les configurations possibles sont décrites dans le tableau 3.1.

Seulement 23 transitions correspondent à une ou plusieurs aptitudes. En effet la transition $t_{2 \rightarrow 1}$, ne correspond qu'à un relâchement qui n'est pas caractéristique d'un mouvement.

3.3 Modélisation de la difficulté d'un exercice

3.3.1 Calibration

La calibration est une étape à part, effectuée en dehors d'un exercice et à chaque début de séance. Elle a pour but d'initialiser le système avec les aptitudes du patient, quoique seulement l'amplitude soit mesurée à l'heure actuelle.

La tâche de calibration demande au patient d'effectuer un mouvement de pronation-supination du poignet pour atteindre son maximum, depuis sa position de repos. La position de repos peut être paume ou dos contre la table, selon la préférence du patient. La position de repos est définie comme la zone où le patient ne fournit aucun effort pour maintenir la position.

La calibration en amplitude terminée, le jeu prend l'amplitude maximale enregistrée comme référence de la colonne extrême du patient.

3.3.2 Principe

Afin de proposer en permanence les exercices les plus adaptés aux performances du patient, il est nécessaire de pouvoir quantifier la difficulté globale de chaque exercice.

Pour cela, le modèle quantifie d'abord la difficulté à chaque ligne, puis en fait la moyenne. La difficulté de la ligne k est la somme de deux composantes :

- la difficulté d'aptitudes : quantifie la difficulté de la transition par addition des coefficients de la configuration.
- la difficulté de l'endurance : décrit par une unique formule la difficulté liée aux notions de maintien, de direction du mouvement et de nombre de mouvements réalisés.

Cette somme est normalisée dans $[0; 1]$ par rapport aux difficultés minimale et maximale possibles pour cette ligne.

3.3.3 Difficulté d'aptitudes

Lors d'une transition de la ligne k à la ligne $k + 1$, le patient fournit un effort particulier par le changement (de la colonne i à la colonne j) ou le maintien (colonne i) de l'orientation de sa main. Les thérapeutes ont tenté de quantifier la difficulté de chaque configuration, par un classement. Les configurations sont composées d'une combinaison d'aptitudes, qui chacune ont un coefficient propre. Ces coefficients sont présentés dans le tableau 3.2.

Pour quantifier la difficulté d'une configuration, on effectue la somme des coefficients associés à chaque aptitude. Ce coefficient est ensuite rapporté entre 0 et 1.

$d_{i \rightarrow j}$ décrit la difficulté de la configuration due à la transition de la colonne i à la colonne j :

Soient i et j deux entiers $\in [1; 5]$ et $E = \{(i, j) \in [1; 5]^2 \setminus (1, 1)\}$

$$d_{i \rightarrow j} = v_{i \rightarrow j} + a_{i \rightarrow j} + m_{i \rightarrow j} + p_{i \rightarrow j} + r_{i \rightarrow j} \quad (3.3)$$

Avec :

$$v_{i \rightarrow j} = \begin{cases} V2 & \text{si } |i - j| = 2 \\ V3 & \text{si } |i - j| = 3 \\ V4 & \text{si } |i - j| = 4 \\ 0 & \text{sinon} \end{cases} \quad a_{i \rightarrow j} = \begin{cases} A4 & \text{si } j = 4 \\ A5 & \text{si } j = 5 \\ 0 & \text{sinon} \end{cases}$$

$$m_{i \rightarrow j} = \begin{cases} M & \text{si } i = j \\ 0 & \text{sinon} \end{cases} \quad p_{i \rightarrow j} = \begin{cases} P & \text{si } j \in [2; 4] \wedge i \neq j \\ 0 & \text{sinon} \end{cases} \quad r_{i \rightarrow j} = \begin{cases} R & \text{si } j = 1 \\ 0 & \text{sinon} \end{cases}$$

Difficulté des aptitudes			
$V2$	0.5	$A4$	1
$V3$	1	$A5$	4
$V4$	2	M	2
P	0.5	R	-0.75

TABLE 3.2 – Coefficients caractérisant la difficulté des aptitudes

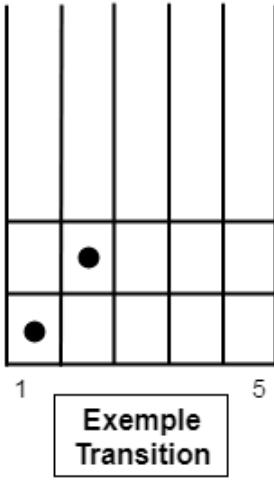
$D_{i \rightarrow j}$ décrit la difficulté de la configuration, normalisée par les difficultés minimale et maximale possibles parmi toutes les configurations depuis la position i.

$$D_{i \rightarrow j} = \frac{d_{i \rightarrow j} - d_{min}}{d_{max} - d_{min}} \quad (3.4)$$

$$\begin{aligned} d_{min} &= \min\{(i, j) \in E \mid d_{i \rightarrow j}\} \\ d_{max} &= \max\{(i, j) \in E \mid d_{i \rightarrow j}\} \end{aligned}$$

Exemple

L'exemple ci-contre (Fig. 3.3) illustre le calcul de la difficulté d'une transition de la colonne 1 en ligne k à la colonne 2 en ligne $k + 1$.



- Vitesse $v_{1 \rightarrow 2} = 0$ car $|j - i| = 1$
- Amplitude $a_{1 \rightarrow 2} = 0$ car $j \neq 4 \wedge 5$
- Maintien $m_{1 \rightarrow 2} = 0$ car $i \neq j$
- Précision $p_{1 \rightarrow 2} = 0.5$ car $j \in [2; 4] \wedge i \neq j$
- Relâchement $r_{1 \rightarrow 2} = 0$ car $j \neq 1$
- Transition (somme) $d_{1 \rightarrow 2} = 0.5$

Les valeurs maximale et minimale sont :

$$d_{min} = d_{2 \rightarrow 1} = -0.75 ; d_{max} = d_{1 \rightarrow 5} = d_{5 \rightarrow 5} = 6$$

FIGURE 3.3 – Transition $t_{1 \rightarrow 2}$

On a donc :

$$D_{1 \rightarrow 2} = \frac{0.5 + 0.75}{6 + 0.75} = 0.185$$

La difficulté de la configuration pour passer de la ligne 1 à la ligne 2 est donc de 0.185.

Changement de direction du mouvement	
hold	0
same	0.25
opposite	0.8

TABLE 3.3 – Coefficients de changement de direction du mouvement

3.3.4 Difficulté de l'endurance

La première composante, la difficulté d'aptitudes, décrit la difficulté intrinsèque d'une configuration d'aptitudes. Or, la difficulté d'un exercice est fortement liée à la succession des pièces, à l'alternance des mouvements et à la fatigue qui en découle.

Un seconde composante a donc été introduite pour décrire cette difficulté de l'endurance, en particulier les cas de maintien de position de la main et les cas de changement de direction.

La fonction qui décrit la difficulté de l'endurance est une fonction de trois variables (h pour *hold*, d pour *direction* et m pour *move*).

- $h_j \in \mathbb{R}^{+*}$ est le nombre de pièces successives présentes dans une même colonne : il s'agit du cas du maintien. La première pièce est omise, car elle n'est pas le résultat d'un maintien mais de la configuration précédente, i.e. le maintien commence à partir de la deuxième cible consécutive dans la même colonne.
- $d_j \in \{\text{hold} ; \text{same} ; \text{opposite}\}$ est le coefficient attribué au changement de direction du mouvement. "hold" indique que l'on se situe en cas de maintien, "same" indique que le mouvement suit la même direction qu'à la pièce précédente, et "opposite" indique un changement de direction depuis la précédente pièce. Le détail des valeurs attribuées à ces coefficients est affiché dans le tableau 3.3. Ces valeurs peuvent être amenées à changer pour modifier le comportement du modèle, au même titre que les coefficients caractérisant la difficulté d'aptitudes.
- $m_j \in \mathbb{R}^{+*}$ est le nombre de changements de direction.

La difficulté de l'endurance est alors :

$$diff_{endurance}(h_j, d_j, m_j) = \begin{cases} H(h_j) & \text{si } d_j = 0 \\ F(m_j, d_j) & \text{si } d_j \neq 0 \end{cases} \quad (3.5)$$

Avec :

$$\begin{aligned} H(h_j) &= 6.10^{-5}h_j^3 + 3.4.10^{-3}h_j^2 + 4.81.10^{-2}h_j + 0.246 \\ F(m_j, d_j) &= 5.10^{-5}m_j^3 + 3.10^{-3}m_j^2 + 4.8.10^{-2}m_j + d_j \end{aligned}$$

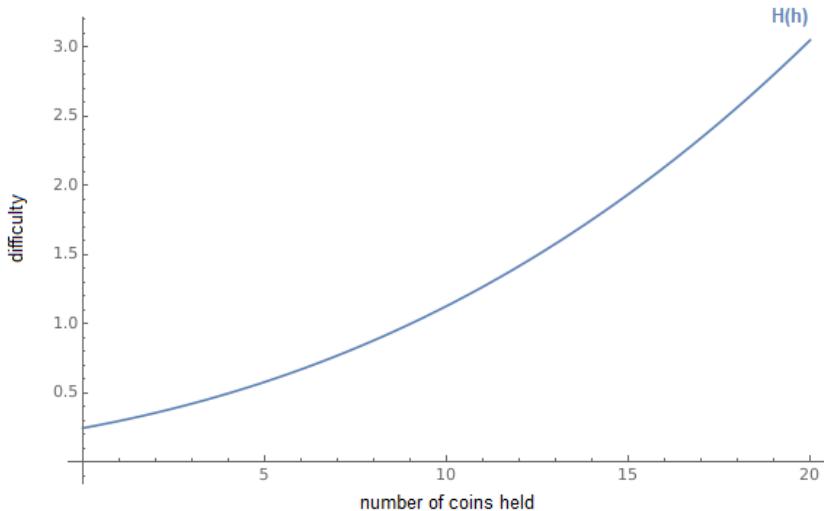


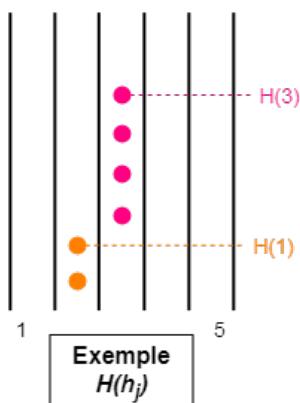
FIGURE 3.4 – Tracé de la fonction $H(h_j)$

Les polynômes $H(h_j)$ et $F(m_j, d_j)$ ont été définis expérimentalement par les ergothérapeutes et le service Recherche de la Clinique "Les Trois Soleils", qui participent à la conception de ce modèle. Ces polynômes ont été créés par approche inverse, en partant d'un résultat envisagé pour le patient et remontant jusqu'aux variables étudiées (Fig. 3.4 et 3.5).

$H(h_j)$ exprime la difficulté d'un maintien de la position en fonction du nombre de pièces successives dans la même colonne. $F(m_j, d_j)$ exprime la difficulté d'une suite de mouvements.

Exemple

Les deux exemples présentés ci-dessous illustrent comment les pièces sont comptabilisées pour les composantes de l'endurance.



Exemple $H(h_j)$: deux maintiens sont effectués. Le premier, en orange, a deux pièces. La première pièce marque l'arrivée dans une nouvelle colonne. Elle ne compte donc pas pour du maintien, puisque rien n'est maintenu à ce stade. On commence donc à compter à partir de la pièce suivante, d'où $H(1)$ pour la deuxième pièce.

Le second maintien, en rose, a quatre pièces. De même, la quatrième pièce correspond à $H(3)$, car la première pièce est simplement le début de la nouvelle suite, par changement de colonne.

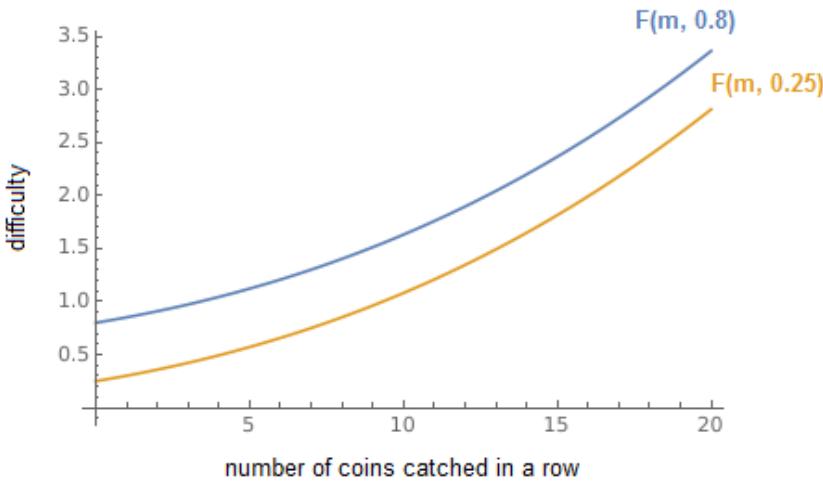
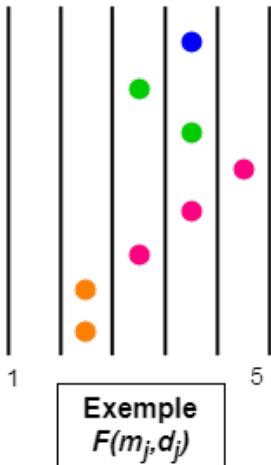


FIGURE 3.5 – Tracé de la fonction $F(m_j, d_j)$



Le polynôme $F(m_j, d_j)$ décrit un autre aspect du mouvement.

Cet aspect tient compte de l’alternance entre mouvements de pronation et de supination. On considère qu’un changement de direction coûte plus d’efforts qu’une poursuite du mouvement. Plus il y a de changements de direction dans un exercice, plus celui-ci est difficile.

Exemple $F(m_j, d_j)$: les différentes couleurs indiquent les différents mouvements réalisés.

- En orange, pas de mouvement car maintien en colonne 2. Sens = 0
- En rose, 1er mouvement. Sens = 0.8 pour la pièce 1, puis 0.25 pour les pièces 2 et 3
- En vert, 2e mouvement. Sens = 0.8 pour la pièce 1, et 0.25 pour la pièce 2
- En bleu, 3e mouvement. Sens = 0.8

Au total, il y eut 3 mouvements.

3.3.5 Difficulté globale

Le calcul de la difficulté globale de l'exercice se divise en deux étapes :

- calcul de la difficulté de la ligne,
- moyenne des difficultés des lignes.

Difficulté de la ligne

La difficulté $g_{i \rightarrow j}$ dépeint la difficulté associée à la transition de la colonne i , en ligne k , vers la colonne j , en ligne $k + 1$, en prenant en compte la difficulté de la configuration de cette transition, ainsi que la difficulté relative au temps en termes de mouvement ou de maintien, c'est à dire la difficulté de l'endurance.

$$g_{i \rightarrow j} = D_{i \rightarrow j} + \text{diff}_{\text{endurance}}(h_j, d_j, m_j) \quad (3.6)$$

Elle est ensuite normalisée afin d'obtenir un indice de difficulté final toujours compris entre 0 et 1 pour prendre en compte les difficultés maximale et minimale possibles à la ligne considérée.

$$g_{i \rightarrow jN} = \frac{g_{i \rightarrow j} - g_{i \rightarrow j\min}}{g_{i \rightarrow j\max} - g_{i \rightarrow j\min}} \quad (3.7)$$

Les difficultés $g_{i \rightarrow j\min}$ et $g_{i \rightarrow j\max}$ représentent respectivement les difficultés minimale et maximale possibles pour la situation courante, i.e. la transition de la colonne i à la colonne j .

La difficulté normalisée $g_{i \rightarrow jN}$ décrit donc la difficulté globale associée à la transition de la colonne i à la colonne j .

Difficulté globale de l'exercice

Nous faisons enfin une moyenne sur l'ensemble des lignes. l est le nombre de lignes de l'exercice. Dans cette étude sur une série de 20 pièces, cette constante vaut 18. La première et la dernière pièces ne sont pas comptabilisées car leurs positions sont fixées à l'identique pour chaque exercice.

$$G = \frac{\sum_{i=1}^l g_{i \rightarrow jN}}{l} \quad (3.8)$$

3.3.6 Conclusion

Ce modèle de description de la difficulté permet de prendre en compte quatre aptitudes travaillées par le patient : amplitude, maintien, vitesse et précision. Ces aptitudes

sont mises en oeuvre au travers de 16 configurations, qui sont des combinaisons d'une ou plusieurs aptitudes à des niveaux différents d'intensité. Ces aptitudes sont alors travaillées et améliorées lors des exercices, qui consistent en des suites de 20 pièces, sur 20 lignes et 5 colonnes. Le patient doit alors ramasser toutes les pièces.

Le modèle évalue la difficulté de chaque ligne, puis en fait la moyenne. La difficulté d'une ligne est la somme de deux composantes : la difficulté d'aptitudes, qui est la somme des coefficients de la configuration de la ligne et la difficulté de l'endurance, qui intègre la difficulté liée aux notions de maintien, de direction du mouvement et de nombre de mouvements réalisés. La difficulté d'aptitudes est évaluée par des coefficients les caractérisant. La difficulté d'une configuration est alors obtenue par somme des coefficients des aptitudes constituant la configuration de la ligne.

La difficulté de l'endurance quant à elle se décrit comme une fonction créée par approche inverse par les ergothérapeutes et le service Recherche de la Clinique "Les Trois Soleils". Elle est constituée de deux polynômes : $H(h_j)$ exprime la difficulté d'un maintien de la position en fonction du nombre de pièces successives dans la même colonne et $F(m_j, d_j)$ exprime la difficulté d'une suite de mouvements.

Après normalisation dans $[0; 1]$ de la somme des deux composantes, une moyenne sur l'ensemble des 18 lignes est réalisée, afin d'obtenir la difficulté globale de l'exercice.

Ce modèle, bien que conçu dans le cadre du jeu de l'oiseau, est flexible, notamment par la présence de coefficients affectant son comportement (importance des aptitudes, des mouvements réalisés). On pourrait ainsi l'appliquer sur d'autres exercices de rééducation simplement en changeant les valeurs de ces paramètres.

Nous avons donc un modèle robuste de description de la difficulté. Il s'agit à présent de créer des tâches et des exercices qui correspondent le mieux au patient et à ses capacités. Pour ce faire, nous allons nous intéresser à la génération d'exercices, selon différents critères et contraintes.

3.4 Génération d'exercices

3.4.1 Problématiques

La génération d'exercices consiste en la création d'un ou plusieurs exercices adaptés en difficulté aux capacités du patient, et faisant travailler différentes aptitudes. Dans notre cas, nous disposons d'un modèle de description de la difficulté basé sur un schéma de 20 lignes et 5 colonnes (section 3.2). Plusieurs problématiques se sont révélées au fil de la réflexion.

Tout d'abord, le nombre très important de schémas possibles. En effet, le jeu de l'oiseau consiste à ramasser des pièces. Chaque pièce est placée dans l'une des 5 colonnes, à raison d'une pièce par ligne, sur 20 lignes. Les lignes 1 et 20 sont imposées en colonne 1, ce qui laisse le choix des 5 colonnes pour 18 lignes. Un calcul rapide montre donc qu'il existe $5^{18} \approx 3,8 \cdot 10^{12}$ exercices. Nous pourrions générer tous ces exercices en avance, afin de pouvoir choisir celui qui est le plus approprié à notre situation. Cependant, des contraintes évidentes de coût (puissance, temps de calcul) nous interdisent cette option.

Nous pourrions alors décider de choisir de manière aléatoire un exercice. Le caractère aléatoire permettrait peut-être de garantir une variabilité assez importante des exercices proposés. Toutefois, cette solution n'est pas viable non plus, car nous travaillons avec des personnes atteintes d'incapacités physiques variables d'un jour à l'autre. Il faut garantir que l'exercice proposé corresponde à leurs capacités du jour, établies par la phase de calibration (section 3.3.1).

Il nous faut donc trouver le moyen de générer **quelques** exercices qui **correspondent** à la fois au patient, et aussi aux tâches qui lui sont demandées. Deux idées ont été explorées. La génération par la difficulté (section 3.4.2) permet de générer des exercices en se basant sur une difficulté cible, que l'on tente d'atteindre par un tirage aléatoire pondéré. Cette approche est viable en termes de variabilité des exercices et d'atteinte des objectifs : elle garantit de toujours sélectionner les exercices les plus pertinents pour le patient. Il s'agit de l'approche choisie pour implémentation dans le jeu de l'oiseau.

La génération contrainte quant à elle (section 3.4.3) fut explorée très tôt dans le projet. L'idée est de placer des contraintes de structure des exercices (par exemple, des exercices ne faisant travailler qu'une aptitude en majorité) afin d'orienter la génération. La complexité de l'idée et la diversité des contraintes nous ont empêché d'approfondir la réflexion, quoiqu'elle reste toujours ouverte.

3.4.2 Génération par la difficulté

L'objectif est de générer un exercice, de difficulté $D \in [0; 1]$ voulue et fournie en argument. La méthode suivie est décrite par l'exemple présenté en figure 3.6.

La ligne k possède une pièce en colonne i . Pour la ligne suivante $k+1$, il y a 5 possibilités : le système pourrait choisir chacune des 5 colonnes. Le système va alors sélectionner une colonne de manière aléatoire pondérée. Ce comportement se réitère pour toutes les lignes jusqu'à l'avant-dernière.

Pour chaque possibilité, nous sommes capables de calculer la difficulté résultante G_{k+1} . Cette difficulté résultante sera à une certaine "distance" de la difficulté cible D .

Nous calculons ensuite la probabilité $P(j)$ de sélectionner une colonne j , pour chaque

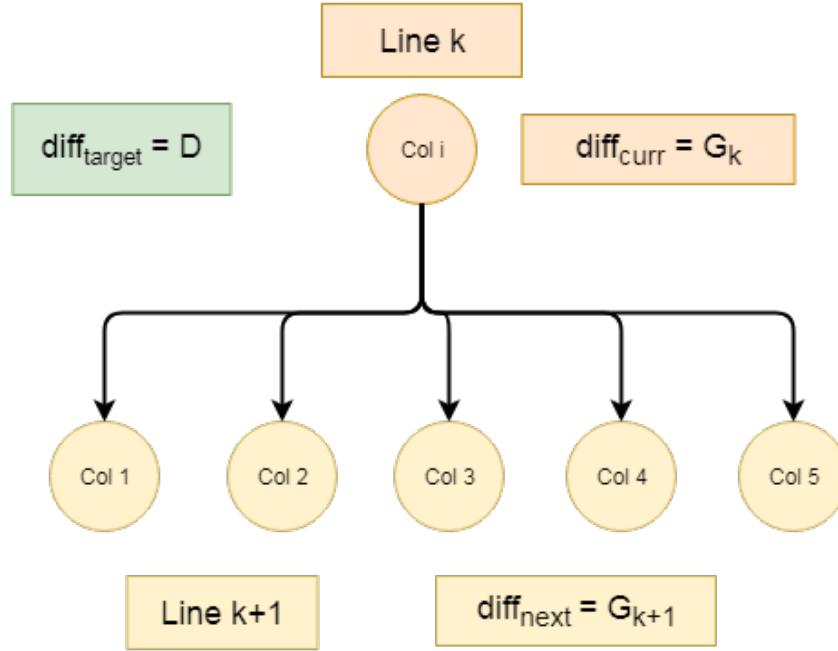


FIGURE 3.6 – Choix de la colonne en ligne $k + 1$

colonne. Ce calcul met en valeur la difficulté résultante la plus proche de la difficulté cible D , sans toutefois interdire de sélectionner les autres possibilités.

$$P(j) = \frac{p_j}{\sum_{j=1}^5 p_j} \quad ; \quad p_j = e^{(1-|G_{k+1}-D|)*10} \quad (3.9)$$

On a donc 5 choix, avec chacun une difficulté résultante et une probabilité de tirage propres. Un tirage aléatoire est enfin réalisé, tenant compte du calcul de probabilité fait ci-dessus. La difficulté courante est ensuite mise à jour.

Ce modèle pourra être ensuite utilisé avec des contraintes supplémentaires pour orienter la génération vers des exercices ciblant des aptitudes en particulier.

Répartition des exercices

Les exercices générés sont répartis entre 11 catégories de difficulté, de la difficulté la plus faible 0 à la difficulté la plus élevée 10. Chaque catégorie a une largeur de 0.1, exceptées les difficultés extrêmes 0 et 10, car n'étant pas souvent atteintes. C'est pourquoi leur largeur n'est que de 0.05. Les catégories sont présentées dans le tableau 3.4.

Catégorie	Intervalle
0	[0 ; 0.05]
1	[0.05 ; 0.15]
2	[0.15 ; 0.25]
3	[0.25 ; 0.35]
4	[0.35 ; 0.45]
5	[0.45 ; 0.55]
6	[0.55 ; 0.65]
7	[0.65 ; 0.75]
8	[0.75 ; 0.85]
9	[0.85 ; 0.95]
10	[0.95 ; 1]

TABLE 3.4 – Correspondance entre catégories et intervalles de difficulté

Lors de la génération, le système crée plusieurs milliers d'exercices, tous ayant la même difficulté cible, mais pas forcément la même difficulté résultante. Le système sélectionne alors l'exercice le plus proche de l'objectif D , afin de répondre au mieux au souhait formulé.

La difficulté cible D fournie en argument peut être affinée, par exemple en fournissant une difficulté cible plus précise. Cependant, le temps de calcul en sera augmenté, et aucune certitude quant à l'atteinte de cette cible n'est garantie.

3.4.3 Exemples de génération contrainte

Introduction

Une idée abordée très tôt dans le projet était de générer des niveaux de tâches selon certaines règles spécifiques, ou contraintes, afin d'orienter le programme de rééducation du patient. Cinq exemples ont pour le moment été explorés, avec la possibilité de les combiner avec la génération par la difficulté décrite dans la sous-section précédente. Les cinq exemples sont :

- Exercices simples, i.e. qui ne font intervenir qu'une seule aptitude par ligne ;
- Exercices complexes, i.e. qui ne font intervenir que des combinaisons d'aptitudes pour chaque ligne ;
- Exercices orientés amplitude : chaque ligne contient majoritairement de l'amplitude ;
- Exercices orientés vitesse : chaque ligne contient majoritairement de la vitesse ;
- Exercices orientés précision : chaque ligne contient majoritairement de la précision.

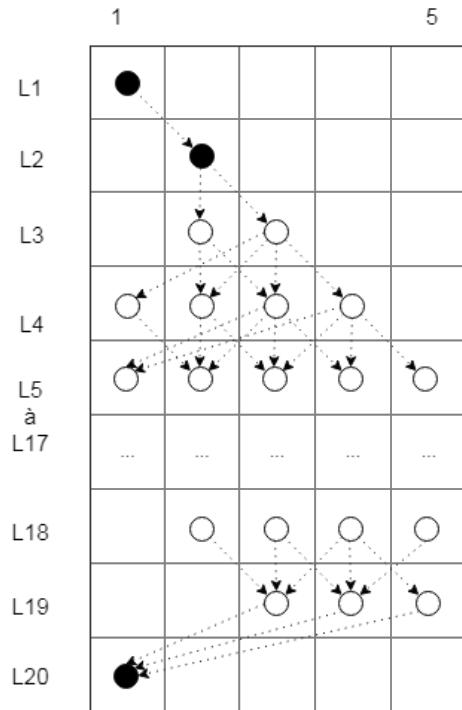


FIGURE 3.7 – Exercices simples

La réflexion faite sur chaque exemple étant très similaire, nous ne présentons que le premier exemple, en décrivant la liste triée par colonne de départ des transitions possibles et l'explication des contraintes appliquées. Pour les autres, nous mentionnons seulement la définition et les éventuels problèmes et limitations.

Les cinq exemples sont des contraintes de génération de type d'exercices, totalement indépendants de la difficulté. Chaque exemple peut générer ensuite des exercices de difficultés variables. Ainsi, il peut y avoir des exercices "simples" plus difficiles que des exercices "complexes". Cette idée n'ayant pas été complètement explorée, elle est encore en discussion au sein de l'équipe et ne sera pas implémentée pour le moment.

Exercices simples

Les exercices simples sont ceux ne faisant intervenir qu'une seule aptitude par ligne, à l'exception de l'amplitude $A4$, qui est forcément associée à du maintien ou de la précision. L'intégration de l'amplitude $A4$ permet de générer des modèles d'exercices utilisant les 5 colonnes. L'exclusion de l'amplitude $A4$ génère des pièces uniquement dans les colonnes 1 à 3.

Les transitions listées par colonne de départ sont :

- 1 : $t_{1 \rightarrow 2}$
- 2 : $t_{2 \rightarrow 2}; t_{2 \rightarrow 3}$
- 3 : $t_{3 \rightarrow 1}; t_{3 \rightarrow 2}; t_{3 \rightarrow 3}; t_{3 \rightarrow 4}$
- 4 : $t_{4 \rightarrow 1}; t_{4 \rightarrow 3}; t_{4 \rightarrow 4}; t_{4 \rightarrow 5}$
- 5 : $t_{5 \rightarrow 1}; t_{5 \rightarrow 4}$

Sur la figure 3.7 illustrant le schéma de construction de cet exemple, les cercles pleins indiquent les positions imposées (uniques), les cercles vides indiquent les positions multiples possibles.

La ligne 2 est imposée par contrainte de la ligne 1 : il n'y a qu'une destination possible depuis la ligne 1. A partir de la ligne 3, on applique les transitions par colonne de départ décrites ci-dessus.

Par exemple, la ligne 2 est en colonne 2, ce qui veut dire qu'il y a 2 transitions possibles pour la ligne 3. La ligne 3 peut être en colonne 2 ou en colonne 3, ce qui veut dire qu'il y a 6 transitions possibles pour la ligne 4. Même analyse pour la ligne 5, avec 11 transitions possibles.

La ligne 19 est imposée par contrainte de la ligne 20, pour utiliser des transitions de la liste. La ligne 18 est imposée par contrainte de la ligne 19. Les lignes 5 à 17 sont totalement libres tant qu'elles respectent les configurations simples possibles définies ci-dessus.

Exercices complexes

Les exercices complexes sont ceux ne faisant intervenir que des combinaisons d'aptitudes. Les transitions d'amplitude *A4* précédemment utilisés pour l'exemple d'exercices simples sont réutilisées ici, car elles sont aussi des combinaisons.

Exercices orientés amplitude

Les exercices orientés amplitude sont ceux où chaque ligne fait travailler au moins l'amplitude. Les lignes peuvent donc contenir de l'amplitude simple, ou des combinaisons d'aptitudes contenant systématiquement de l'amplitude.

Le problème majeur de cet exemple est qu'il interdit l'accès à 3 colonnes sur 5. Il faudrait par exemple :

- inclure des retours ponctuels en colonnes 1, 2 et 3, actuellement inaccessibles, afin d'inclure les transitions associées ;
- redéfinir ce qu'un exercice orienté amplitude signifie. On pourrait changer la formulation en "contenant principalement de l'amplitude" et donc en autorisant la présence d'autres aptitudes.

- réfléchir à une autre méthode ou une autre logique de génération.

Exercices orientés vitesse

Les exercices orientés vitesse sont ceux où chaque ligne fait travailler au moins la vitesse. Les lignes peuvent donc contenir de la vitesse simple, ou des combinaisons d'aptitudes contenant systématiquement de la vitesse.

Exercices orientés précision

Les exercices orientés précision sont ceux où chaque ligne fait travailler au moins la précision. Les lignes peuvent donc contenir de la précision simple, ou des combinaisons d'aptitudes contenant systématiquement de la précision.

En étudiant les transitions associées à cet exemple, nous remarquons qu'il manque de diversité, notamment par l'absence des colonnes 1 et 5. Il serait donc judicieux de permettre l'atteinte de la colonne 5, pour ensuite faire des retours en colonnes 2, 3 et 4. Le risque d'une telle modification est d'inclure trop d'amplitude, caractéristique de la colonne 5, et donc de dénaturer le but des exercices orientés précision.

Exercices orientés maintien

Le maintien est une aptitude un peu à part, car elle ne nécessite pas de transition de colonne pour être travaillée.

Les exercices orientés maintien consistent donc à proposer des suites de pièces plus ou moins longues dans des colonnes fixes, en changeant de temps en temps la colonne. Le changement ne doit pas être trop fréquent, pour ne pas risquer de voir apparaître une autre aptitude. Les maintiens en colonnes 4 et 5 feront systématiquement travailler l'amplitude en plus.

Le seul paramètre à décider est donc celui de la longueur d'une série (moins de 20 pièces), la colonne du maintien étant sélectionnée de manière aléatoire.

3.5 Évaluation et Adaptation

3.5.1 Évaluation de la performance

L'évaluation de la performance du patient fait partie de la dernière phase de la boucle de session. Le système va juger les performances mesurées pendant la phase de complétion des exercices, et produire un score, une note qui déterminera l'adaptation à réaliser.

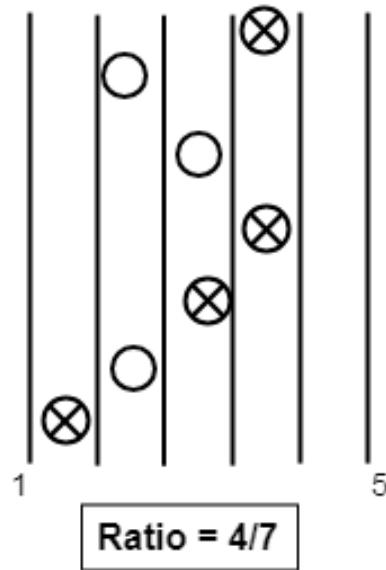


FIGURE 3.8 – Ratio pièces ramassées sur pièces présentées

Un exemple de score calculé est le ratio entre le nombre de pièces que le patient a ramassé et le nombre de pièces qui lui furent présentées (Fig. 3.8). Ce ratio est à l'heure actuelle le seul score utilisé pour l'adaptation. D'autres statistiques peuvent cependant être extraites, pour apporter des feedbacks au patient, concernant sa performance et son évolution, et aussi un suivi de la rééducation pour le thérapeute.

Une autre piste actuellement en cours d'exploration est la quantification de la réussite. Cette quantification consiste à analyser chaque configuration proposée et chaque configuration réalisée par le patient, afin de déduire des scores de réussite par ligne, par configuration, ou par aptitude.

Le système est donc capable, pour tous les types de configuration, de noter la réussite du patient et d'en produire un score de réussite, qui montre les zones où il se sent à l'aise, les zones où il a besoin d'entraînement, y compris si les configurations ne sont pas initialement proposées dans l'exercice.

Un tel outil permet de suivre la progression des patients de manière personnalisée, et d'adapter le programme de rééducation en conséquence.

3.5.2 Adaptation

L'adaptation est la dernière étape de la boucle de session. En se basant sur le ou les scores calculés dans l'étape d'évaluation de la performance, le système décide de la meilleure action à entreprendre pour toujours proposer un exercice adapté aux capacités

du patient.

Dans le cadre du jeu de l'oiseau, nous cherchons à stabiliser le patient dans un intervalle de performance, choisi arbitrairement. Il faut donc que les exercices proposés soient suffisamment accessibles pour permettre la réussite du patient, sans toutefois les rendre trop faciles, afin qu'une progression soit toujours possible. Pour cela, l'intervalle de performance proposé est [60% – 80%].

Si le ratio calculé à la section 3.5.1 est dans l'intervalle, le système lui proposera un nouvel exercice issu de la même catégorie de difficulté. Si le ratio est inférieur à 60%, le système proposera au patient un nouvel exercice issu de la catégorie en deçà de la catégorie actuelle. A l'inverse, si le ratio est supérieur à 80%, le système lui proposera un nouvel exercice issu de la catégorie en dessus de la catégorie actuelle.

A chaque adaptation, le nouvel exercice est généré selon la méthode de génération par la difficulté (voir section 3.4.2).

Au début, une grille de 10 exercices dans chacun des 11 niveaux de difficulté est générée, afin de s'assurer d'avoir une base d'exercices possibles. Cette base d'exercices sera générée à chaque début de séance, et affinée pendant celle-ci.

3.6 Conclusion

Pour ce projet, et en particulier pour le jeu de l'oiseau, le problème était de proposer un exercice adapté et adaptable au patient, qui puisse tenir compte de ses capacités quotidiennes et de plusieurs aptitudes à entraîner, dans le cadre d'un programme de rééducation à domicile.

Pour y répondre, notre solution se base sur un modèle de description de la difficulté, qui prend en compte lesdites aptitudes, que sont l'amplitude, la précision, la vitesse et le maintien. Ces aptitudes forment par combinaison ou non des configurations, pour lesquelles le modèle attribue une notation de difficulté. Ainsi, chaque exercice est décrit configuration par configuration, il est annoté d'une catégorie de difficulté puis classé en 11 catégories.

Ce modèle de description de la difficulté s'inscrit dans un processus en cinq phases (Fig. 3.1). Tout d'abord, une calibration est effectuée à chaque début de séance, afin d'initialiser le système avec les capacités du jour du patient. La tâche réalisée en phase de calibration consiste à atteindre le maximum d'amplitude, afin de paramétrier l'angle de rotation du poignet correspondant à la colonne extrême.

Ensuite, la phase de génération produit un exercice, qui initialement est basé sur le résultat de la calibration, puis sur le résultat de l'exercice précédent.

Cet exercice est complété par le patient : la consigne est simplement de ramasser toutes les pièces présentes. Pendant ce temps, les performances du patient sont mesurées par le système, en temps réel.

Enfin, la phase d'évaluation de la performance et d'adaptation permet de finaliser le processus. Le système note la performance du patient, par calcul du ratio des pièces ramassées sur celles présentées, et décide de l'adaptation à effectuer, en fonction de l'intervalle de réussite fixé.

A l'heure actuelle, il s'agit de l'intervalle [60% – 80%], ce qui signifie que si le ratio est à l'intérieur, aucune adaptation n'est faite et un exercice issu de la même catégorie de difficulté sera généré lors de la boucle de session suivante. Sinon, si le ratio est inférieur à 60% (respectivement, supérieur à 80%), le système générera un exercice issu de la catégorie inférieure (respectivement, catégorie supérieure) de difficulté.

Le modèle de description et d'adaptation de la difficulté permet donc de répondre au problème formulé, en garantissant de garder le patient au centre du sujet, toujours acteur de sa rééducation.

Chapitre 4

Conclusion et perspectives

4.1 Conclusion du stage

Le projet de ce stage était de concevoir, développer et implémenter un processus de difficulté adaptative pour le jeu de l'oiseau, composant du projet plus global "*Cesaar-AVC*", en partenariat avec l'entreprise "*Interaction Healthcare*" et la clinique "*Les Trois Soleils*". A mon arrivée, le jeu de l'oiseau était déjà en cours de développement, et la difficulté proposée était identique pour tous les patients. Le problème consistait donc à produire un processus capable de s'adapter à chaque patient, chaque jour et quelles que soient leurs capacités.

La solution finalement proposée répond autant que possible au problème. Elle repose sur un modèle, qui s'inscrit dans un processus en cinq phases. Ce modèle permet d'aboutir à une description de la difficulté des exercices proposés, qui tient compte de tous les aspects de la rééducation : aptitudes travaillées par le patient et les éventuelles combinaisons de ces aptitudes, durée de l'exercice, et fatigue due au mouvement. Les cinq phases du processus sont clairement identifiables, et s'articulent autour du patient. La calibration récupère ses capacités journalières, la génération d'exercice se base sur ses résultats précédents, les mesures de performance s'effectuent pendant l'exécution des tâches, puis l'évaluation de la performance et l'adaptation lui permettent de s'améliorer sans s'ennuyer ni abandonner.

Au moment de la rédaction de ce rapport, le modèle est arrêté. Son implémentation dans le jeu de l'oiseau a débuté, sous la forme de scripts directement appliqués dans les mécaniques du jeu thérapeutique. Ce modèle fait par ailleurs l'objet d'une publication dans la conférence VRST (*Symposium on Virtual Reality Software and Technology*). Il reste cependant plusieurs points à arranger. Le plus important d'entre eux est l'ajustement des coefficients des aptitudes, qui régissent le comportement du modèle. Ces coefficients permettent de donner une importance différente aux aptitudes travaillées par les patients, et donc de changer l'échelle de difficulté des configurations. Cet aspect étant

critique, il est important d'effectuer plusieurs tests auprès des patients, avec des coefficients différents, afin que la difficulté résultante du modèle corresponde à la difficulté perçue par les thérapeutes et les patients pendant leur rééducation.

4.2 Conclusion personnelle

Au moment de la rédaction de ce rapport (août), mon stage n'est pas encore terminé. Aussi, je conclurai sur mon expérience jusqu'ici, sur ce que le stage m'a apporté, les plus et les moins rencontrés pendant son déroulement, ainsi que sur le bilan de mes attentes.

J'attendais beaucoup de ce stage. Tout d'abord, je voulais avoir un avis réel et approfondi de la recherche, de l'environnement et de la méthode de travail. Ces aspects ont été pleinement satisfaits : la veille constante, afin de toujours être sûr que ce que l'on fait n'a pas déjà été accompli par d'autres personnes, la remise en question permanente, qui permet de jauger et d'évaluer notre propre travail, le mettre en exergue du travail d'autrui, afin de le corriger, ou de l'améliorer. Par ailleurs, la collaboration entre les membres de l'équipe, même si nos sujets sont différents, est un aspect auquel je n'avais pas pensé, et qui m'a beaucoup plu. Je fais notamment référence à une discussion qu'Amine, Romaric, Ludovic, Guillaume et moi avons eu, un vendredi soir de juillet, qui nous a mobilisé trois heures sans que nous voyions le temps passer.

Je ne distingue pas de points négatifs en rapport direct avec le stage qui vaillent la peine d'être mentionnés ici.

Pour les apports, ce stage a complété mon idée et mon expérience précédentes de la recherche. J'ai amélioré mon autonomie, notamment lors de la recherche d'idées, ou l'orientation de mon état de l'art, mais aussi ma communication. Mon stage étant l'une des parties du projet de thèse de Ludovic David, j'ai été amené à beaucoup discuter et travailler avec lui, dans un premier temps pour comprendre l'ampleur et les enjeux du projet, puis par la suite pour élaborer notre modèle, l'affiner, le remettre en question, et ainsi de suite jusqu'à aboutir au modèle décrit dans ce rapport.

4.3 Perspectives

Ce stage s'inscrit dans le cadre du projet "Cesaar-AVC", plus global, sur lequel Ludovic David travaille déjà pour sa thèse de doctorat. Ce projet vise à concevoir et à développer un outil de rééducation post-AVC du membre supérieur à domicile, à l'aide d'un dispositif bas coût.

Dans ce contexte, d'autres problématiques furent formulées, dont une qui m'intéresse au moins autant que le stage réalisé ces derniers mois. Il s'agit de comprendre l'impact des feedbacks sur l'expérience utilisateur, et notamment les effets sur la motivation, le

confort et la performance du patient. Cet aspect fut abordé pendant le stage, mais non exploré car il s'éloignait trop du sujet initial. C'est pourquoi j'ai exprimé l'envie de poursuivre ces travaux dans le cadre d'études de troisième cycle.

Bibliographie

- [1] Robin HUNICKE et Vernell CHAPMAN. « AI for Dynamic Difficulty Adjustment in Games ». In : Proceedings of the Challenges in Game AI Workshop, Nineteenth National Conference on Artificial Intelligence. 2004.
- [2] Minhua MA et Kamal BECHKOUN. « Serious Games for Movement Therapy after Stroke ». In : IEEE International Conference on Systems, Man and Cybernetics. 2008, p. 1872–1877.
- [3] Gazihan ALANKUS et al. « Towards Customizable Games for Stroke Rehabilitation ». In : Proceedings of the 28th International Conference on Human Factors in Computing Systems. 2010, p. 2113–2122.
- [4] Martin JENNINGS-TEATS, Gillian SMITH et Noah WARDRIP-FRUIN. « Polymorph : Dynamic Difficulty Adjustment Through Level Generation ». In : Proceedings of the 2010 Workshop on Procedural Content Generation in Games. 2010, p. 11.
- [5] Jens NIRME, Armin DUFF et Paul F.M.J. VERSCHURE. « Adaptive Rehabilitation Gaming System : On-line individualization of stroke rehabilitation ». In : Annual International Conference of the IEEE Engineering in Medicine and Biology Society. IEEE, 2011.
- [6] Michele PIROVANO et al. « Self-Adaptive Games for Rehabilitation at Home ». In : IEEE Conference on Computational Intelligence and Games (CIG). IEEE, 2012.
- [7] Dayana HRISTOVA. *Dynamic Difficulty Adjustment (DDA) in First Person Shooter (FPS) Games*. 2014.
- [8] Yi-Na LI et al. « Adaptive Difficulty Scales for Parkour Games ». In : *Journal of Visual Languages & Computing* 25.6 (2014), p. 868–878.
- [9] Ana de los REYES-GUZMÁN et al. « Quantitative assessment based on kinematic measures of functional impairments during upper extremity movements : A review ». In : *Clinical Biomechanics* 29.7 (2014), p. 719–727. DOI : <http://dx.doi.org/10.1016/j.clinbiomech.2014.06.013>. (Visité le 28/03/2018).
- [10] Nadia HOCINE et al. « Adaptation in serious games for upper-limb rehabilitation : An approach to improve training outcomes ». In : *User Modeling and User-Adapted Interaction* 25.1 (2015), p. 65–98.

- [11] Rhio SUTOYO et al. « Dynamic Difficulty Adjustment in Tower Defence ». In : *Procedia Computer Science* 59 (2015), p. 435–444.
- [12] Koji MIKAMI, Kondo KUNIO et Henry FERNANDEZ. « Adaptable Game Experience Based on Player's Performance and EEG ». In : Nicograph International. 2017.