

IFT609 – INFORMATIQUE COGNITIVE

Rapport de projet



Université de Sherbrooke

Cohen Johana, De La Purification Sandra, Pécault Léo, Rebiha Farah, Vauthier Aurélien

Table des matières

1. Introduction.....	2
1.1 Domaine	2
1.2 Problématique.....	2
2. Représentation des connaissances	2
2.1 Buts.....	2
2.1.1 But principal	2
2.1.2 Sous-buts	2
2.2 Types de chunks	3
2.2.1 La texture.....	3
2.2.2 L'expérience de fouettage.....	3
2.3 Procédures.....	3
2.3.1 startCassage.....	3
2.3.2 attraperOeuf.....	4
2.3.3 casserOeufSansCoquille ou casserOeufAvecCoquille	4
2.3.4 separerOeufSansJaune ou separerOeufAvecJaune.....	4
2.3.5 verifierPasCoquille.....	4
2.3.6 verifierAvecCoquille, retirerCoquille puis retirerCoquilleTermine.....	4
2.3.7 verifierPasJaune	4
2.3.8 verifierAvecJaune, retirerJaune et retirerJauneTermine	4
2.3.9 passerAuFouettage puis demarrerFouettage	4
2.3.10 recupererVitesseSucces.....	4
2.3.11 recupererVitesseSuccesOK ou recupererVitesseSuccesKO.....	5
2.3.12 recupererVitesseManqueEnergieOK ou recupererVitesseManqueEnergieKO.....	5
2.3.13 baisserVitesse ou augmenterVitesse.....	5
2.3.14 fouetterClassique	5
2.3.15 verifierTexture puis comparerTextureOK ou comparerTextureKO.....	5
2.3.16 verifierEnergieOK ou verifierEnergieKO	5
2.3.17 finirExperienceEchec ou finirExperienceSucces	5
3. Exécution du programme.....	6
3.1 Scénario simulé	6
3.2 Trace	6
4. Apprentissage.....	12
4.1 Casser et séparer les œufs	12
4.2 Monter les blancs en neige	13
5. Conclusion	14

1. Introduction

1.1 Domaine

Nous avons choisi d’aborder le thème de la cuisine pour ce projet. En effet, c’est non seulement une activité qui nous unit tous mais également qui fait partie intégrante de notre quotidien. Dans le vaste univers culinaire, nous avons spécifiquement sélectionné le fait de monter les blancs d’œufs en neige car c’est une tâche en apparence simple mais qui est, finalement, loin d’être évidente à maîtriser pour la réussir à tous les coups.

1.2 Problématique

« Monter les blancs en neige » est donc la tâche que nous avons choisi de modéliser. L’idée est d’obtenir une texture parfaite de blanc d’œufs en neige, c’est-à-dire qui soit ferme, non liquide, avec une couleur proche du blanc et un aspect très aéré. Nous pourrions alors jouer sur plusieurs paramètres tels que la vitesse pour fouetter les blancs, le geste utilisé et le temps passé à fouetter. Afin qu’il soit le plus proche possible d’un comportement humain, notre modèle ne pourra pas fouetter à une vitesse trop élevée ni pendant trop longtemps. C’est pourquoi nous avons décidé de prendre en compte la fatigue accumulée au cours du temps.

Nous avons également défini des hypothèses de départ, que l’on va considérer comme étant admises pour la suite, ce qui évitera de trop complexifier notre modèle :

- Tous les ingrédients et ustensiles sont à disposition
- Le plan de travail est en place, prêt à être utilisé
- Notre modèle sait compter

2. Représentation des connaissances

2.1 Buts

2.1.1 But principal

Le but principal de notre modèle est d’obtenir des blancs en neige. Pour l’atteindre, celui-ci doit réaliser deux sous-buts : n’avoir que des blancs d’œufs et les battre jusqu’à obtention de la texture souhaitée. Notre modèle, au départ, dispose d’un certain nombre d’œufs à battre ainsi que d’une réserve d’énergie qui va décroître au fil du temps. Si la texture parfaite n’est pas obtenue avant que l’énergie ne tombe à 0, le modèle échoue à atteindre son but principal et s’arrête.

But principal :

```
(chunk-type butAvoirBlancsEnNeige statut)
```

2.1.2 Sous-buts

Notre premier sous-but est l’obtention de blancs d’œufs uniquement. Lors de ce sous-but, il va non seulement falloir casser les œufs dans le récipient mais également les séparer pour ne conserver que le blanc. Lors de cette opération, il est probable que certains résidus de coquilles et de jaune se

retrouvent mêlés à nos blancs, il faut donc prendre en compte, si besoin, le nettoyage du mélange avant de passer au sous-but suivant.

Premier sous-but :

```
(chunk-type butAvoirBlancs nbBlancs nbJaunes nbCoquilles statut)
```

Une fois ce sous-but réalisé, nos blancs sont en place dans le récipient et nettoyés d'éventuels résidus de coquilles ou de jaune. Notre modèle va pouvoir les battre pour obtenir l'aspect en neige. Obtenir une texture en neige est donc notre second sous-but. Différents paramètres vont alors entrer en jeu pour prendre en compte la vitesse d'exécution, le geste réalisé et le temps passé à fouetter. C'est à partir de ce sous-but que l'énergie disponible dans le but principal va commencer à décroître.

Second sous-but :

```
(chunk-type butObtenirTexture couleurActuelle volumeActuel vitesseActuelle energie statut)
```

2.2 Types de chunks

2.2.1 La texture

Ce chunk texture correspond à la texture parfaite de blancs en neige que l'on souhaite atteindre. Elle est caractérisée par une couleur proche du blanc, une certaine consistance ainsi qu'une densité. Ces deux derniers paramètres nous permettent d'indiquer les critères d'une texture ferme et aérée.

Texture :

```
(chunk-type texture couleur volume)
```

2.2.2 L'expérience de fouettage

L'expérience de fouettage enregistre toutes les tentatives de notre modèle pour monter les blancs en neige lors d'un run. C'est ce chunk qui va lui permettre un apprentissage, en essayant à chaque tentative de tendre vers le chunk texture, énoncé précédemment comme la texture parfaite à obtenir. Pour modéliser cela, nous avons choisi d'enregistrer la texture obtenue par notre modèle en fonction de la couleur actuelle, du volume actuel mais aussi de la vitesse d'exécution et du résultat obtenu.

Expérience :

```
(chunk-type experienceFouettage couleurActuelle volumeActuel vitesseMouvement resultat)
```

2.3 Procédures

2.3.1 startCassage

Cette procédure va permettre d'amorcer le cassage des œufs lors du démarrage du programme si *butAvoirBlancs* est le but sélectionné et si son statut est *startCassage*. Elle déclenche donc la recherche d'œuf à casser.

2.3.2 attraperOeuf

Avant de casser des œufs, le modèle va visuellement les chercher dans la fenêtre graphique. Une fois l'œuf trouvé, il place son curseur dessus et change le statut du goal sur *oeufAttrape* pour simuler le geste de prendre l'œuf et le casser dans le récipient.

2.3.3 casserOeufSansCoquille ou casserOeufAvecCoquille

L'une ou l'autre de ces procédures s'exécute selon l'habitude du modèle : s'il est débutant en cassage d'œufs, le risque de mettre de la coquille dans la préparation est plus grand qu'une fois le geste répété et acquis. Cela nous permet de faire un premier apprentissage via l'habitude en récompensant le modèle lorsqu'il ne met pas de coquille dans la solution. Ainsi, s'il rate son geste, c'est *casserOeufAvecCoquille* qui s'exécute pour mettre des coquilles dans la préparation et *casserOeufSansCoquille* sinon. Dans tous les cas, il passe ensuite à la séparation du jaune et du blanc.

2.3.4 separerOeufSansJaune ou separerOeufAvecJaune

Le principe de ces deux procédures est le même que pour les coquilles.

2.3.5 verifierPasCoquille

Cette procédure va permettre au modèle de savoir s'il y a, ou non, des coquilles dans la préparation. S'il n'y en a pas, il va pouvoir vérifier la présence de jaune.

2.3.6 verifierAvecCoquille, retirerCoquille puis retirerCoquilleTermine

La procédure *verifierAvecCoquille* va indiquer au modèle qu'il y a des coquilles dans la préparation et qu'il faut donc la nettoyer pour les retirer avant de passer à la vérification de la présence de jaune. C'est ainsi que *retirerCoquille* s'enclenche, permettant donc de retirer toutes les coquilles. La procédure *retirerCoquilleTermine* va ensuite permettre de passer à la vérification des jaunes.

2.3.7 verifierPasJaune

Cette procédure teste la non-présence de jaune dans la solution et permet de recommencer le processus de cassage d'œufs.

2.3.8 verifierAvecJaune, retirerJaune et retirerJauneTermine

Le principe de ces procédures est le même que pour les coquilles.

2.3.9 passerAuFouettage puis demarrerFouettage

Cette procédure teste s'il n'y a plus aucun œuf à casser pour ensuite commencer le processus de fouettage. Concrètement, elle change le but en cours de traitement de *goalAvoirBlancs* à *goalObtenirTexture*.

2.3.10 recupererVitesseSucces

Permet de récupérer la vitesse d'une précédente tentative terminée par un succès.

2.3.11 *recupererVitesseSuccesOK* ou *recupererVitesseSuccesKO*

Dans un premier temps, le modèle va essayer de se rappeler d'une expérience passée où il a réussi à obtenir une texture parfaite de blancs en neige. S'il y arrive, il récupère la vitesse de fouettage qu'il avait utilisée. Sinon, il cherchera une expérience ratée.

2.3.12 *recupererVitesseManqueEnergieOK* ou *recupererVitesseManqueEnergieKO*

Dans le cas où le modèle n'arrive pas à se rappeler une expérience où il a réussi à obtenir des blancs en neige corrects, il va chercher une expérience passée où il n'a pas réussi. La procédure *recupererVitesseManqueEnergieOK* va s'exécuter s'il arrive à se rappeler d'une précédente expérience où il a échoué. Il va alors récupérer la vitesse qu'il avait appliqué et chercher à la baisser pour réduire sa dépense d'énergie et maximiser ses chances de réussite. Dans le cas où il n'a aucune expérience, le modèle va choisir une vitesse aléatoire comme premier essai.

2.3.13 *baisserVitesse* ou *augmenterVitesse*

En fonction du succès de l'expérience récupérée, le modèle va choisir de diminuer la vitesse rappelée ou de l'augmenter pour optimiser au mieux son énergie.

2.3.14 *fouetterClassique*

Cette procédure représente le fouettage réalisé par le modèle pour monter ses blancs en neige. Elle permet de mettre à jour au fur et à mesure le volume et la couleur de la préparation en fonction de la vitesse d'exécution tout en diminuant l'énergie encore disponible. Ainsi, la texture s'améliore à chaque itération mais, en contrepartie, l'énergie du modèle diminue.

2.3.15 *verifierTexture* puis *comparerTextureOK* ou *comparerTextureKO*

La procédure *verifierTexture* permet au modèle de se rappeler la texture parfaite enregistrée dans la base de connaissances. Ainsi, il va pouvoir se servir de cette information récupérée pour la comparer avec celle qu'il a actuellement dans son récipient. Si cette texture parfaite a été atteinte, le modèle arrête de fouetter et va enregistrer la tentative comme étant un succès. Sinon, il va chercher à continuer de fouetter pour l'atteindre mais devra d'abord s'assurer d'avoir assez d'énergie pour continuer.

2.3.16 *verifierEnergieOK* ou *verifierEnergieKO*

Ces procédures permettent au modèle de savoir s'il lui reste encore de l'énergie ou non. Si c'est le cas, il continue le fouettage en exécutant *fouetterClassique*. Sinon, il arrête et selon la valeur de la texture obtenue, la tentative sera un succès ou un échec.

2.3.17 *finirExperienceEchec* ou *finirExperienceSucces*

Si le modèle a réussi à monter les blancs en neige, *finirExperienceSucces* sera la procédure exécutée. Elle va permettre d'enregistrer cette tentative comme un succès pour que notre modèle puisse apprendre et s'en rappeler plus tard. En revanche, s'il n'a pas réussi, c'est *finirExperienceEchec* qui s'exécutera pour enregistrer cette expérience comme étant ratée. L'enregistrement des valeurs de

couleur, de volume, de vitesse d'exécution et de résultat (échec ou succès) permettront alors au modèle d'adapter son geste pour les itérations futures.

3. Exécution du programme

3.1 Scénario simulé

Le scénario que nous présentons dans ce rapport est le cas dans lequel le modèle n'a aucune expérience pour casser, séparer ou fouetter des œufs. Il va donc commencer par chercher les œufs, les attraper puis les casser avec une certaine probabilité de mettre des coquilles dans la solution. Puis, il va séparer le blanc du jaune d'œuf avec une certaine probabilité de mettre un peu de jaune dans la solution. Une fois ces étapes faites pour un œuf, il vérifie si des coquilles ou du jaune auraient été ajoutés à la solution et si oui, il retire les éléments non désirés. Il va répéter ce processus tant qu'il y aura des œufs dans son champ de vision.

Une fois ces étapes de cassage et de séparation effectuées pour tous les œufs, le modèle va passer au fouettage. Dans un premier temps, il va essayer de se rappeler d'une précédente expérience où sa tentative de fouettage des œufs a réussi. S'il y arrive, il utilisera cette vitesse pour fouetter les œufs. S'il ne se rappelle pas d'une tentative réussie, il va chercher une tentative où il n'a pas réussi à obtenir des blancs en neige avant de s'épuiser. Dans ce cas-là, il va réduire sa vitesse pour essayer de moins dépenser d'énergie pour atteindre une texture en neige. S'il n'a absolument aucune expérience, il va prendre une valeur de vitesse aléatoire.

3.2 Trace

(experiment)		
0.000	GOAL	SET-BUFFER-CHUNK GOAL GOALAVOIRBLANCSENNEIGE REQUESTED NIL
0.000	GOAL	SET-BUFFER-CHUNK GOAL BUTAVOIRBLANCSENNEIGE0 REQUESTED NIL
0.000	VISION	SET-BUFFER-CHUNK VISUAL-LOCATION VISUAL-LOCATION0-0
REQUESTED NIL		
0.000	PROCEDURAL	CONFLICT-RESOLUTION
0.050	PROCEDURAL	PRODUCTION-FIRED START
0.050	GOAL	SET-BUFFER-CHUNK GOAL GOALAVOIRBLANCS REQUESTED NIL
0.050	PROCEDURAL	CONFLICT-RESOLUTION
0.100	PROCEDURAL	PRODUCTION-FIRED STARTCASSAGE
0.100	PROCEDURAL	CLEAR-BUFFER VISUAL-LOCATION
0.100	VISION	Find-location
0.100	VISION	SET-BUFFER-CHUNK VISUAL-LOCATION VISUAL-LOCATION2-0
0.100	PROCEDURAL	CONFLICT-RESOLUTION
0.150	PROCEDURAL	PRODUCTION-FIRED ATTRAPEROEUF
0.150	PROCEDURAL	CLEAR-BUFFER VISUAL
0.150	PROCEDURAL	CLEAR-BUFFER MANUAL
0.150	MOTOR	MOVE-CURSOR LOC VISUAL-LOCATION2-0-0
0.150	PROCEDURAL	CONFLICT-RESOLUTION
0.235	VISION	Encoding-complete VISUAL-LOCATION2-0-0 NIL
0.235	VISION	SET-BUFFER-CHUNK VISUAL TEXT0
0.235	PROCEDURAL	CONFLICT-RESOLUTION
0.350	PROCEDURAL	CONFLICT-RESOLUTION
0.400	PROCEDURAL	CONFLICT-RESOLUTION
0.581	PROCEDURAL	CONFLICT-RESOLUTION
0.631	PROCEDURAL	CONFLICT-RESOLUTION
0.681	PROCEDURAL	PRODUCTION-FIRED CASSEROEUFVECCOQUILLE
1		
0.681	PROCEDURAL	CLEAR-BUFFER MANUAL
0.681	MOTOR	CLICK-MOUSE
0.681	UTILITY	PROPAGATE-REWARD 1

```

Utility updates with Reward = 1.0   alpha = 0.2
Updating utility of production START
  U(n-1) = 0.0   R(n) = 0.319 [1.0 - 0.681 seconds since selection]
  U(n) = 0.0638
Updating utility of production STARTCASSAGE
  U(n-1) = 0.0   R(n) = 0.36900002 [1.0 - 0.631 seconds since selection]
  U(n) = 0.073800005
Updating utility of production ATTRAPEROEUF
  U(n-1) = 0.0   R(n) = 0.41900003 [1.0 - 0.581 seconds since selection]
  U(n) = 0.08380001
Updating utility of production CASSEROEUFVECCOQUILLE
  U(n-1) = 0.0   R(n) = 0.95 [1.0 - 0.05 seconds since selection]
  U(n) = 0.19
    0.681   PROCEDURAL           CONFLICT-RESOLUTION
    0.731   PROCEDURAL           PRODUCTION-FIRED SEPAREROEUFVECCJAU
    0.731   UTILITY              PROPAGATE-REWARD 1
Utility updates with Reward = 1.0   alpha = 0.2
Updating utility of production SEPAREROEUFVECCJAU
  U(n-1) = 0.0   R(n) = 0.95 [1.0 - 0.05 seconds since selection]
  U(n) = 0.19
    0.731   PROCEDURAL           CONFLICT-RESOLUTION
    0.781   PROCEDURAL           PRODUCTION-FIRED VERIFIERVECCOQUILLE
    0.781   PROCEDURAL           CONFLICT-RESOLUTION
    0.831   PROCEDURAL           PRODUCTION-FIRED RETIRERCOQUILLE
    0.831   PROCEDURAL           CONFLICT-RESOLUTION
    0.881   PROCEDURAL           PRODUCTION-FIRED RETIRERCOQUILLETERMINE
    0.881   PROCEDURAL           CONFLICT-RESOLUTION
    0.891   MOTOR                OUTPUT-KEY #(28 2)
    0.931   PROCEDURAL           PRODUCTION-FIRED VERIFIERVECCJAU
    0.931   PROCEDURAL           CLEAR-BUFFER VISUAL
    0.931   PROCEDURAL           CLEAR-BUFFER MANUAL
    0.931   PROCEDURAL           CONFLICT-RESOLUTION
    0.976   VISION               Encoding-complete VISUAL-LOCATION2-0-0 NIL
    0.976   VISION               No visual-object found
    0.981   PROCEDURAL           PRODUCTION-FIRED RETIRERJAU
    0.981   PROCEDURAL           CONFLICT-RESOLUTION
    1.031   PROCEDURAL           PRODUCTION-FIRED RETIRERJAUNETERMINE
    1.031   PROCEDURAL           CONFLICT-RESOLUTION
    1.081   PROCEDURAL           PRODUCTION-FIRED STARTCASSAGE
    1.081   PROCEDURAL           CLEAR-BUFFER VISUAL-LOCATION
    1.081   VISION               Find-location
    1.081   VISION               SET-BUFFER-CHUNK VISUAL-LOCATION VISUAL-LOCATION4-1
    1.081   PROCEDURAL           CONFLICT-RESOLUTION
    1.131   PROCEDURAL           PRODUCTION-FIRED ATTRAPEROEUF
    1.131   PROCEDURAL           CLEAR-BUFFER VISUAL
    1.131   PROCEDURAL           CLEAR-BUFFER MANUAL
    1.131   MOTOR                MOVE-CURSOR LOC VISUAL-LOCATION4-1-0
    1.131   PROCEDURAL           CONFLICT-RESOLUTION
    1.216   VISION               Encoding-complete VISUAL-LOCATION4-1-0 NIL
    1.216   VISION               SET-BUFFER-CHUNK VISUAL TEXT1
    1.216   PROCEDURAL           CONFLICT-RESOLUTION
    1.331   PROCEDURAL           CONFLICT-RESOLUTION
    1.381   PROCEDURAL           CONFLICT-RESOLUTION
    1.493   PROCEDURAL           CONFLICT-RESOLUTION
    1.543   PROCEDURAL           CONFLICT-RESOLUTION
    1.593   PROCEDURAL           PRODUCTION-FIRED CASSEROEUFVECCOQUILLE
1
    1.593   PROCEDURAL           CLEAR-BUFFER MANUAL
    1.593   MOTOR                CLICK-MOUSE
    1.593   UTILITY              PROPAGATE-REWARD 1
Utility updates with Reward = 1.0   alpha = 0.2
Updating utility of production VERIFIERVECCOQUILLE
  U(n-1) = 0.0   R(n) = 0.13800001 [1.0 - 0.862 seconds since selection]
  U(n) = 0.027600003
Updating utility of production RETIRERCOQUILLE
  U(n-1) = 0.0   R(n) = 0.18800002 [1.0 - 0.812 seconds since selection]
  U(n) = 0.037600007
Updating utility of production RETIRERCOQUILLETERMINE

```



```

U(n-1) = 0.0   R(n) = 0.23799998 [1.0 - 0.762 seconds since selection]
U(n) = 0.04759997
Updating utility of production VERIFIERAVECJAUNE
U(n-1) = 0.0   R(n) = 0.288 [1.0 - 0.712 seconds since selection]
U(n) = 0.0576
Updating utility of production RETIRERJAUNE
U(n-1) = 0.0   R(n) = 0.338 [1.0 - 0.662 seconds since selection]
U(n) = 0.067600004
Updating utility of production RETIRERJAUNETERMINE
U(n-1) = 0.0   R(n) = 0.388 [1.0 - 0.612 seconds since selection]
U(n) = 0.0776
Updating utility of production STARTCASSAGE
U(n-1) = 0.073800005   R(n) = 0.43800002 [1.0 - 0.562 seconds since selection]
U(n) = 0.14664
Updating utility of production ATTRAPEROEUF
U(n-1) = 0.08380001   R(n) = 0.48799998 [1.0 - 0.512 seconds since selection]
U(n) = 0.16464001
Updating utility of production CASSEROEUFVECCOQUILLE
U(n-1) = 0.19   R(n) = 0.95 [1.0 - 0.05 seconds since selection]
U(n) = 0.342
1.593   PROCEDURAL           CONFLICT-RESOLUTION
1.643   PROCEDURAL           PRODUCTION-FIRED SEPAREROEUFVECJAUNE
1.643   UTILITY              PROPAGATE-REWARD 1
Utility updates with Reward = 1.0   alpha = 0.2
Updating utility of production SEPAREROEUFVECJAUNE
U(n-1) = 0.19   R(n) = 0.95 [1.0 - 0.05 seconds since selection]
U(n) = 0.342
1.643   PROCEDURAL           CONFLICT-RESOLUTION
1.693   PROCEDURAL           PRODUCTION-FIRED VERIFIERAVECCOQUILLE
1.693   PROCEDURAL           CONFLICT-RESOLUTION
1.743   PROCEDURAL           PRODUCTION-FIRED RETIRERCOQUILLE
1.743   PROCEDURAL           CONFLICT-RESOLUTION
1.793   PROCEDURAL           PRODUCTION-FIRED RETIRERCOQUILLETERMINE
1.793   PROCEDURAL           CONFLICT-RESOLUTION
1.803   MOTOR                OUTPUT-KEY #(28 2)
1.843   PROCEDURAL           PRODUCTION-FIRED VERIFIERAVECJAUNE
1.843   PROCEDURAL           CLEAR-BUFFER VISUAL
1.843   PROCEDURAL           CLEAR-BUFFER MANUAL
1.843   PROCEDURAL           CONFLICT-RESOLUTION
1.888   VISION               Encoding-complete VISUAL-LOCATION4-1-0 NIL
1.888   VISION               No visual-object found
1.893   PROCEDURAL           PRODUCTION-FIRED RETIRERJAUNE
1.893   PROCEDURAL           CONFLICT-RESOLUTION
1.943   PROCEDURAL           PRODUCTION-FIRED RETIRERJAUNETERMINE
1.943   PROCEDURAL           CONFLICT-RESOLUTION
1.993   PROCEDURAL           PRODUCTION-FIRED STARTCASSAGE
1.993   PROCEDURAL           CLEAR-BUFFER VISUAL-LOCATION
1.993   VISION               Find-location
1.993   VISION               SET-BUFFER-CHUNK VISUAL-LOCATION VISUAL-LOCATION0-2
1.993   PROCEDURAL           CONFLICT-RESOLUTION
2.043   PROCEDURAL           PRODUCTION-FIRED ATTRAPEROEUF
2.043   PROCEDURAL           CLEAR-BUFFER VISUAL
2.043   PROCEDURAL           CLEAR-BUFFER MANUAL
2.043   MOTOR                MOVE-CURSOR LOC VISUAL-LOCATION0-2-0
2.043   PROCEDURAL           CONFLICT-RESOLUTION
2.128   VISION               Encoding-complete VISUAL-LOCATION0-2-0 NIL
2.128   VISION               SET-BUFFER-CHUNK VISUAL TEXT2
2.128   PROCEDURAL           CONFLICT-RESOLUTION
2.243   PROCEDURAL           CONFLICT-RESOLUTION
2.293   PROCEDURAL           CONFLICT-RESOLUTION
2.487   PROCEDURAL           CONFLICT-RESOLUTION
2.537   PROCEDURAL           CONFLICT-RESOLUTION
2.587   PROCEDURAL           PRODUCTION-FIRED CASSEROEUFVECCOQUILLE
1
2.587   PROCEDURAL           CLEAR-BUFFER MANUAL
2.587   MOTOR                CLICK-MOUSE
2.587   UTILITY              PROPAGATE-REWARD 1
Utility updates with Reward = 1.0   alpha = 0.2

```

```

Updating utility of production VERIFIERAVECCOQUILLE
U(n-1) = 0.027600003 R(n) = 0.055999994 [1.0 - 0.944 seconds since selection]
U(n) = 0.03328
Updating utility of production RETIRERCOQUILLE
U(n-1) = 0.037600007 R(n) = 0.106000006 [1.0 - 0.894 seconds since selection]
U(n) = 0.051280007
Updating utility of production RETIRERCOQUILLETERMINE
U(n-1) = 0.047599997 R(n) = 0.15600002 [1.0 - 0.844 seconds since selection]
U(n) = 0.06928
Updating utility of production VERIFIERAVECJAUNE
U(n-1) = 0.0576 R(n) = 0.20599997 [1.0 - 0.794 seconds since selection]
U(n) = 0.08728
Updating utility of production RETIRERJAUNE
U(n-1) = 0.067600004 R(n) = 0.25599998 [1.0 - 0.744 seconds since selection]
U(n) = 0.10528
Updating utility of production RETIRERJAUNETERMINE
U(n-1) = 0.0776 R(n) = 0.306 [1.0 - 0.694 seconds since selection]
U(n) = 0.12328
Updating utility of production STARTCASSAGE
U(n-1) = 0.14664 R(n) = 0.356 [1.0 - 0.644 seconds since selection]
U(n) = 0.188512
Updating utility of production ATTRAPEROEUF
U(n-1) = 0.16464001 R(n) = 0.40600002 [1.0 - 0.594 seconds since selection]
U(n) = 0.21291201
Updating utility of production CASSEROEUFVECCOQUILLE
U(n-1) = 0.342 R(n) = 0.95 [1.0 - 0.05 seconds since selection]
U(n) = 0.4636
    2.587 PROCEDURAL CONFLICT-RESOLUTION
    2.637 PROCEDURAL PRODUCTION-FIRED SEPAREROEUFVECCJAUNE
    2.637 UTILITY PROPAGATE-REWARD 1
Utility updates with Reward = 1.0 alpha = 0.2
Updating utility of production SEPAREROEUFVECCJAUNE
U(n-1) = 0.342 R(n) = 0.95 [1.0 - 0.05 seconds since selection]
U(n) = 0.4636
    2.637 PROCEDURAL CONFLICT-RESOLUTION
    2.687 PROCEDURAL PRODUCTION-FIRED VERIFIERAVECCOQUILLE
    2.687 PROCEDURAL CONFLICT-RESOLUTION
    2.737 PROCEDURAL PRODUCTION-FIRED RETIRERCOQUILLE
    2.737 PROCEDURAL CONFLICT-RESOLUTION
    2.787 PROCEDURAL PRODUCTION-FIRED RETIRERCOQUILLETERMINE
    2.787 PROCEDURAL CONFLICT-RESOLUTION
    2.797 MOTOR OUTPUT-KEY #(28 2)
    2.837 PROCEDURAL PRODUCTION-FIRED VERIFIERAVECJAUNE
    2.837 PROCEDURAL CLEAR-BUFFER VISUAL
    2.837 PROCEDURAL CLEAR-BUFFER MANUAL
    2.837 PROCEDURAL CONFLICT-RESOLUTION
    2.882 VISION Encoding-complete VISUAL-LOCATION0-2-0 NIL
    2.882 VISION No visual-object found
    2.887 PROCEDURAL PRODUCTION-FIRED RETIRERJAUNE
    2.887 PROCEDURAL CONFLICT-RESOLUTION
    2.937 PROCEDURAL PRODUCTION-FIRED RETIRERJAUNETERMINE
    2.937 PROCEDURAL CONFLICT-RESOLUTION
    2.987 PROCEDURAL PRODUCTION-FIRED STARTCASSAGE
    2.987 PROCEDURAL CLEAR-BUFFER VISUAL-LOCATION
    2.987 VISION Find-location
    2.987 VISION SET-BUFFER-CHUNK VISUAL-LOCATION VISUAL-LOCATION6-3
    2.987 PROCEDURAL CONFLICT-RESOLUTION
    3.037 PROCEDURAL PRODUCTION-FIRED ATTRAPEROEUF
    3.037 PROCEDURAL CLEAR-BUFFER VISUAL
    3.037 PROCEDURAL CLEAR-BUFFER MANUAL
    3.037 MOTOR MOVE-CURSOR LOC VISUAL-LOCATION6-3-0
    3.037 PROCEDURAL CONFLICT-RESOLUTION
    3.122 VISION Encoding-complete VISUAL-LOCATION6-3-0 NIL
    3.122 VISION SET-BUFFER-CHUNK VISUAL TEXT3
    3.122 PROCEDURAL CONFLICT-RESOLUTION
    3.237 PROCEDURAL CONFLICT-RESOLUTION
    3.287 PROCEDURAL CONFLICT-RESOLUTION
    3.533 PROCEDURAL CONFLICT-RESOLUTION

```

```

3.583 PROCEDURAL CONFLICT-RESOLUTION
3.633 PROCEDURAL PRODUCTION-FIRED CASSEROEUFVECCOQUILLE
1
3.633 PROCEDURAL CLEAR-BUFFER MANUAL
3.633 MOTOR CLICK-MOUSE
3.633 UTILITY PROPAGATE-REWARD 1
Utility updates with Reward = 1.0 alpha = 0.2
Updating utility of production VERIFIERAVECCOQUILLE
U(n-1) = 0.03328 R(n) = 0.004000008 [1.0 - 0.996 seconds since selection]
U(n) = 0.027424002
Updating utility of production RETIRERCOQUILLE
U(n-1) = 0.051280007 R(n) = 0.05400002 [1.0 - 0.946 seconds since selection]
U(n) = 0.05182401
Updating utility of production RETIRERCOQUILLETERMINE
U(n-1) = 0.06928 R(n) = 0.10399997 [1.0 - 0.896 seconds since selection]
U(n) = 0.07622399
Updating utility of production VERIFIERAVECJAUNE
U(n-1) = 0.08728 R(n) = 0.15399998 [1.0 - 0.846 seconds since selection]
U(n) = 0.100623995
Updating utility of production RETIRERJAUNE
U(n-1) = 0.10528 R(n) = 0.204 [1.0 - 0.796 seconds since selection]
U(n) = 0.12502399
Updating utility of production RETIRERJAUNETERMINE
U(n-1) = 0.12328 R(n) = 0.254 [1.0 - 0.746 seconds since selection]
U(n) = 0.149424
Updating utility of production STARTCASSAGE
U(n-1) = 0.188512 R(n) = 0.30400002 [1.0 - 0.696 seconds since selection]
U(n) = 0.2116096
Updating utility of production ATTRAPEROEUF
U(n-1) = 0.21291201 R(n) = 0.35399997 [1.0 - 0.646 seconds since selection]
U(n) = 0.2411296
Updating utility of production CASSEROEUFVECCOQUILLE
U(n-1) = 0.4636 R(n) = 0.95 [1.0 - 0.05 seconds since selection]
U(n) = 0.56088
3.633 PROCEDURAL CONFLICT-RESOLUTION
3.683 PROCEDURAL PRODUCTION-FIRED SEPAREROEUFSAJSJAUNE
3.683 UTILITY PROPAGATE-REWARD 3
Utility updates with Reward = 3.0 alpha = 0.2
Updating utility of production SEPAREROEUFSAJSJAUNE
U(n-1) = 0.0 R(n) = 2.95 [3.0 - 0.05 seconds since selection]
U(n) = 0.59000003
3.683 PROCEDURAL CONFLICT-RESOLUTION
3.733 PROCEDURAL PRODUCTION-FIRED VERIFIERAVECCOQUILLE
3.733 PROCEDURAL CONFLICT-RESOLUTION
3.783 PROCEDURAL PRODUCTION-FIRED RETIRERCOQUILLE
3.783 PROCEDURAL CONFLICT-RESOLUTION
3.833 PROCEDURAL PRODUCTION-FIRED RETIRERCOQUILLETERMINE
3.833 PROCEDURAL CONFLICT-RESOLUTION
3.843 MOTOR OUTPUT-KEY #(28 2)
3.883 PROCEDURAL PRODUCTION-FIRED VERIFIERPASJAUNE
3.883 PROCEDURAL CLEAR-BUFFER VISUAL
3.883 PROCEDURAL CLEAR-BUFFER MANUAL
3.883 PROCEDURAL CONFLICT-RESOLUTION
3.928 VISION Encoding-complete VISUAL-LOCATION6-3-0 NIL
3.928 VISION No visual-object found
3.933 PROCEDURAL PRODUCTION-FIRED STARTCASSAGE
3.933 PROCEDURAL CLEAR-BUFFER VISUAL-LOCATION
3.933 VISION Find-location
3.933 VISION FIND-LOC-FAILURE
3.933 PROCEDURAL CONFLICT-RESOLUTION
3.983 PROCEDURAL PRODUCTION-FIRED PASSERAUFUETTAGE
3.983 GOAL SET-BUFFER-CHUNK GOAL GOALAVOIRBLANCSENNEIGE REQUESTED NIL
3.983 PROCEDURAL CLEAR-BUFFER VISUAL-LOCATION
3.983 PROCEDURAL CONFLICT-RESOLUTION
4.033 PROCEDURAL PRODUCTION-FIRED DEMARRERFOUETTAGE
4.033 GOAL SET-BUFFER-CHUNK GOAL GOALOBTENIRTEXTURE REQUESTED NIL
4.033 PROCEDURAL CLEAR-BUFFER VISUAL
4.033 PROCEDURAL CLEAR-BUFFER MANUAL

```

4.033	PROCEDURAL	CLEAR-BUFFER VISUAL-LOCATION
4.033	PROCEDURAL	CONFLICT-RESOLUTION
4.083	PROCEDURAL	PRODUCTION-FIRED RECUPERERVITESSESUCCESS
4.083	PROCEDURAL	CLEAR-BUFFER RETRIEVAL
4.083	DECLARATIVE	START-RETRIEVAL
4.083	DECLARATIVE	RETRIEVAL-FAILURE
4.083	PROCEDURAL	CONFLICT-RESOLUTION
4.133	PROCEDURAL	PRODUCTION-FIRED RECUPERERVITESSESUCCESSKO
4.133	PROCEDURAL	CLEAR-BUFFER RETRIEVAL
4.133	DECLARATIVE	START-RETRIEVAL
4.133	DECLARATIVE	RETRIEVAL-FAILURE
4.133	PROCEDURAL	CONFLICT-RESOLUTION
4.183	PROCEDURAL	PRODUCTION-FIRED RECUPERERVITESSEMANQUEENERGIEKO
4.183	PROCEDURAL	CLEAR-BUFFER RETRIEVAL
4.183	PROCEDURAL	CONFLICT-RESOLUTION
4.233	PROCEDURAL	PRODUCTION-FIRED FOUETTERCLASSIQUE
4.233	PROCEDURAL	CONFLICT-RESOLUTION
4.283	PROCEDURAL	PRODUCTION-FIRED VERIFIERTEXTURE
4.283	PROCEDURAL	CLEAR-BUFFER RETRIEVAL
4.283	DECLARATIVE	START-RETRIEVAL
4.283	DECLARATIVE	RETRIEVED-CHUNK TEXTUREPARFAITE
4.283	DECLARATIVE	SET-BUFFER-CHUNK RETRIEVAL TEXTUREPARFAITE
4.283	PROCEDURAL	CONFLICT-RESOLUTION
4.333	PROCEDURAL	PRODUCTION-FIRED COMPARERTEXTUREKO
4.333	PROCEDURAL	CLEAR-BUFFER RETRIEVAL
4.333	PROCEDURAL	CONFLICT-RESOLUTION
4.383	PROCEDURAL	PRODUCTION-FIRED VERIFIERENERGIEOK
4.383	PROCEDURAL	CONFLICT-RESOLUTION
4.433	PROCEDURAL	PRODUCTION-FIRED FOUETTERCLASSIQUE
4.433	PROCEDURAL	CONFLICT-RESOLUTION
4.483	PROCEDURAL	PRODUCTION-FIRED VERIFIERTEXTURE
4.483	PROCEDURAL	CLEAR-BUFFER RETRIEVAL
4.483	DECLARATIVE	START-RETRIEVAL
4.483	DECLARATIVE	RETRIEVED-CHUNK TEXTUREPARFAITE
4.483	DECLARATIVE	SET-BUFFER-CHUNK RETRIEVAL TEXTUREPARFAITE
4.483	PROCEDURAL	CONFLICT-RESOLUTION
4.533	PROCEDURAL	PRODUCTION-FIRED COMPARERTEXTUREKO
4.533	PROCEDURAL	CLEAR-BUFFER RETRIEVAL
4.533	PROCEDURAL	CONFLICT-RESOLUTION
4.583	PROCEDURAL	PRODUCTION-FIRED VERIFIERENERGIEOK
4.583	PROCEDURAL	CONFLICT-RESOLUTION
4.633	PROCEDURAL	PRODUCTION-FIRED FOUETTERCLASSIQUE
4.633	PROCEDURAL	CONFLICT-RESOLUTION
4.683	PROCEDURAL	PRODUCTION-FIRED VERIFIERTEXTURE
4.683	PROCEDURAL	CLEAR-BUFFER RETRIEVAL
4.683	DECLARATIVE	START-RETRIEVAL
4.683	DECLARATIVE	RETRIEVED-CHUNK TEXTUREPARFAITE
4.683	DECLARATIVE	SET-BUFFER-CHUNK RETRIEVAL TEXTUREPARFAITE
4.683	PROCEDURAL	CONFLICT-RESOLUTION
4.733	PROCEDURAL	PRODUCTION-FIRED COMPARERTEXTUREKO
4.733	PROCEDURAL	CLEAR-BUFFER RETRIEVAL
4.733	PROCEDURAL	CONFLICT-RESOLUTION
4.783	PROCEDURAL	PRODUCTION-FIRED VERIFIERENERGIEOK
4.783	PROCEDURAL	CONFLICT-RESOLUTION
4.833	PROCEDURAL	PRODUCTION-FIRED FOUETTERCLASSIQUE
4.833	PROCEDURAL	CONFLICT-RESOLUTION
4.883	PROCEDURAL	PRODUCTION-FIRED VERIFIERTEXTURE
4.883	PROCEDURAL	CLEAR-BUFFER RETRIEVAL
4.883	DECLARATIVE	START-RETRIEVAL
4.883	DECLARATIVE	RETRIEVED-CHUNK TEXTUREPARFAITE
4.883	DECLARATIVE	SET-BUFFER-CHUNK RETRIEVAL TEXTUREPARFAITE
4.883	PROCEDURAL	CONFLICT-RESOLUTION
4.933	PROCEDURAL	PRODUCTION-FIRED COMPARERTEXTUREKO
4.933	PROCEDURAL	CLEAR-BUFFER RETRIEVAL
4.933	PROCEDURAL	CONFLICT-RESOLUTION
4.983	PROCEDURAL	PRODUCTION-FIRED VERIFIERENERGIEKO
4.983	GOAL	SET-BUFFER-CHUNK GOAL GOALAVOIRBLANCSENNEIGE REQUESTED NIL
4.983	PROCEDURAL	CLEAR-BUFFER IMAGINAL

4.983	PROCEDURAL	CONFLICT-RESOLUTION
5.183	IMAGINAL	SET-BUFFER-CHUNK IMAGINAL CHUNK0
5.183	PROCEDURAL	CONFLICT-RESOLUTION
5.233	PROCEDURAL	PRODUCTION-FIRED FINIREXPERIENCEECHEC
5.233	PROCEDURAL	CLEAR-BUFFER IMAGINAL
5.233	PROCEDURAL	CONFLICT-RESOLUTION
10.000	-----	Stopped because time limit reached

(1 3/4 0)

4. Apprentissage

4.1 Casser et séparer les œufs

Afin de permettre à notre modèle de s'améliorer et d'apprendre de ses erreurs, nous avons mis en place deux méthodes d'apprentissage. La première est basée sur les habitudes et le principe en ACT-R de *reward*. En effet, pour que le comportement de notre modèle soit le plus humain possible, nous avons décidé qu'il serait possible qu'il fasse tomber des coquilles ou du jaune d'œuf dans la préparation au moment où il les casse et les sépare. Cette possibilité se traduit par des procédures possédant les mêmes préconditions, l'une ajoutant une coquille ou du jaune et l'autre non. Lorsqu'une de ces procédures est utilisée, un *reward* lui est appliqué dans le but de favoriser les procédures ne rajoutant pas de coquille ou de jaune dans la solution. Nous pouvons observer cela sur les graphiques ci-dessous, où 25% sur l'axe des ordonnées représente qu'en moyenne les modèles ont déposé 1 coquille ou 1 jaune au cours de l'itération donnée.

Nous avons aussi cherché à ce que cet apprentissage soit optimal après 5 itérations (soit 20 œufs cassés et séparés), tout en gardant toujours une probabilité non nulle d'ajouter des coquilles et du jaune car comme les humains, notre modèle n'est pas infallible. Nous considérons en effet qu'après avoir cassé et séparé une vingtaine d'œufs, une personne ordinaire a appris à ne presque plus faire tomber de coquille ou de jaune. Nous avons cependant pris la décision de garder une probabilité non nulle de mal casser ou de mal séparer le jaune du blanc car nous pensons aussi qu'il arrive de temps en temps d'être maladroit.

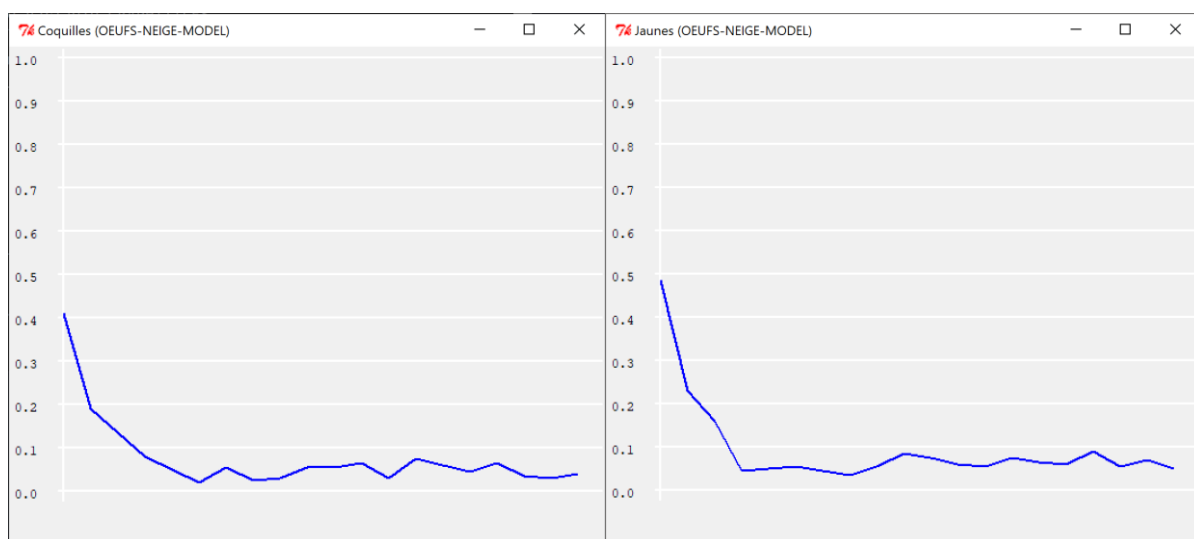


Figure 1 - Courbe d'apprentissage représentant le dépôt de coquille sur 20 itérations (80 cassage d'œufs)

Figure 2 - Courbe d'apprentissage représentant le dépôt de jaune sur 20 itérations (80 cassages d'œufs)

4.2 Monter les blancs en neige

Concernant la montée des blancs en neige, nous avons choisi de nous inspirer du Lab 4 en utilisant un chunk d'apprentissage. Le chunk *experienceFouettage* permet ainsi au modèle de se souvenir de ses tentatives passées afin de progresser vers la réussite du but principal. Cette progression est réalisée dans ACT-R grâce au rappel d'une précédente expérience et à la modification de la vitesse utilisée précédemment. La génération de la vitesse par défaut, c'est-à-dire la vitesse utilisée par le modèle au cours de sa première itération, utilise deux paramètres ayant pour but, d'une part, de permettre à deux exécutions indépendantes du modèle d'être différentes, et d'autre part, de procurer une certaine flexibilité d'apprentissage au modèle : le premier paramètre est la vitesse minimale permettant en pratique de définir une vitesse par défaut minimale mais aussi de définir le point de départ pour l'apprentissage du modèle alors que le deuxième paramètre est la marge aléatoire qui représente la taille de l'intervalle dans lequel nous tirons un nombre aléatoire que nous ajoutons à la vitesse minimale. Cela permet ainsi d'avoir une vitesse par défaut propre à chaque modèle.

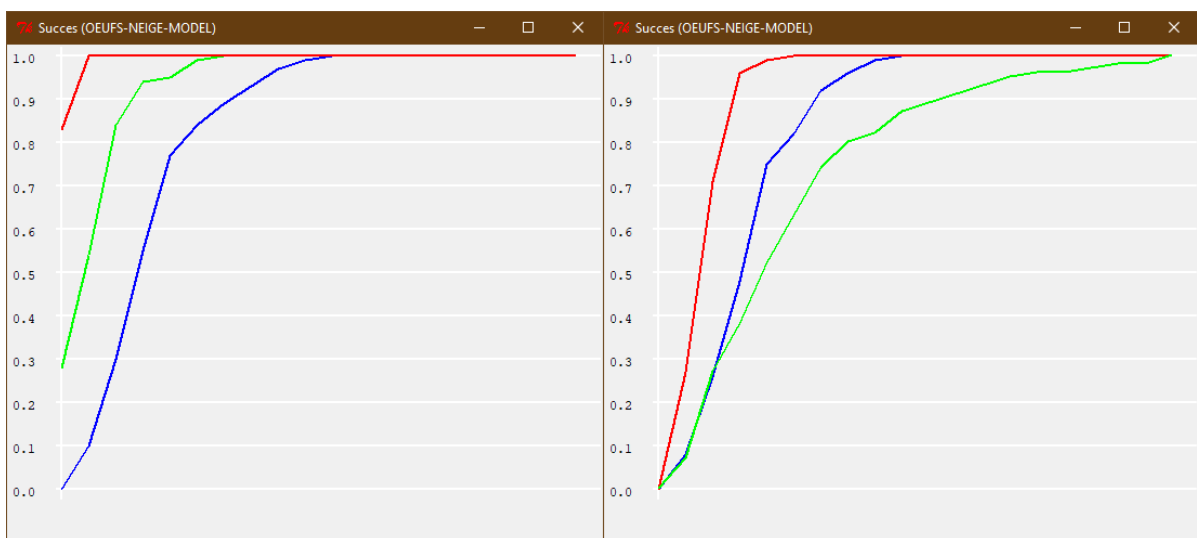


Figure 3 - Courbe d'apprentissage représentant la montée des blancs en neige au fil des itérations en modifiant la vitesse minimale

Figure 1 - Courbe d'apprentissage représentant la montée des blancs en neige au fil des itérations en modifiant la marge aléatoire

Les figures 3 et 4 ci-dessus illustrent donc les différentes courbes d'apprentissage en fonction de la valeur des paramètres fixée. Sur la figure 3, la courbe de référence, en bleu, avec une vitesse minimale de 15, nous indique qu'aucun modèle n'a réussi à monter les blancs en neige au cours de la première itération. En revanche, la courbe verte correspondant à une vitesse de 10 et la rouge à une vitesse de 5, montrent que moins de 10% des modèles n'ont pas réussi à obtenir de blancs en neige à la première itération. D'un autre côté, la modification de la marge aléatoire sur la figure 4 influe sur la progression de l'apprentissage des modèles. La courbe bleue représente, là encore, la courbe de référence avec les mêmes paramètres que la courbe bleue du précédent graphique. Nous voyons ainsi qu'en réduisant la marge aléatoire (courbe rouge), le modèle semble devenir plus performant à la tâche. A contrario, la courbe verte nous montre qu'en réduisant la marge aléatoire, le modèle a plus de difficulté à arriver aux 100% de réussite à monter les blancs en neige.

5. Conclusion

La réalisation de ce modèle montant des blancs d'œufs en neige n'a pas été chose aisée mais nous avons réussi à développer la totalité des objectifs que nous nous étions fixés au départ : notre modèle est en mesure de casser des œufs, séparer le blanc du jaune et fouetter les blancs jusqu'à l'obtention d'une texture en neige. Nous avons jugé intéressant d'ajouter un second type d'apprentissage, par l'habitude, pour les étapes de cassage et de séparation des œufs bien qu'il ne soit pas initialement prévu. En effet, il nous a permis de réutiliser les compétences acquises du Lab 5, ce qui complète parfaitement celles du Lab 4 que nous utilisons pour l'apprentissage du fouettage. Les résultats obtenus simulent un apprentissage plutôt cohérent avec la réalité : ils sont proches de ceux qu'un être humain pourrait obtenir.

Notre modèle n'est, toutefois, pas parfait. En effet, les paramètres fixés pour représenter la vitesse de fouettage ainsi que la couleur et le volume de la texture restent assez abstraits : ils ne sont pas basés sur des expériences que nous aurions faites en situation réelle. Le modèle mathématique choisi pour représenter le geste de fouettage manque également de réalisme. Il serait donc intéressant de mener des expériences pour mesurer ces données et modéliser au mieux la situation.

Bien que nous soyons satisfaits de notre modèle, nous avons réfléchi à d'éventuelles améliorations pour une évolution future. Au-delà de corriger les défauts de modélisation cités plus haut, nous pourrions également ajouter la possibilité d'augmenter temporairement la vitesse de fouettage, permettant au modèle de réaliser un « rush » dans un dernier espoir d'avoir la texture si celle-ci ne monte pas. Il serait également intéressant de prendre en compte un éventuel échec lors du nettoyage des jaunes, ce qui aurait un impact sur la montée des blancs. Enfin, il serait plus réaliste d'adapter l'énergie du modèle selon la forme physique de notre modèle car il se peut qu'il ne puisse pas réussir si un jour il est trop fatigué.

Finalement, notre projet est adaptable à une multitude de situations en cuisine. Il est possible de le décliner pour toutes les recettes nécessitant de casser des œufs, séparer le jaune et le blanc et fouetter la préparation telles que blanchir des jaunes, réaliser une meringue et bien d'autres.