

# TREX BRAIN

## Etat d'avancement

### Remise du contexte

Pour rappel, le projet consiste à mettre en place une IA qui apprend d'elle-même à jouer au jeu du T-Rex de Google Chrome (Easter Egg caché lorsque l'on ouvre le navigateur Chrome sans connexion internet).

Le projet se décompose en 3 étapes :

1. D'abord, coder le jeu en javascript,
2. Ensuite, mettre en place le réseau neuronal associé à un T-Rex,
3. Enfin, faire évoluer le réseau neuronal afin que l'IA fasse un score maximal.



*Capture d'écran du jeu inclus dans le navigateur Chrome*

### Travail réalisé

#### Codage du jeu

Actuellement, le jeu a été codé en Javascript. Il est jouable par un humain. Le joueur doit faire sauter le T-Rex lorsqu'il rencontre un cactus, ou bien l'accroupir lorsqu'il rencontre un ptérodactyle. La librairie p5 a été utilisée pour gérer facilement la boucle de jeu (update et draw). Cependant, cette librairie est assez contraignante et me pose des problèmes lorsque j'essaie de mettre en place l'algorithme génétique... Je n'ai pas pris le jeu existant car je ne pouvais pas accéder aux informations relatives aux neurones d'entrée.

#### Mise en place d'un réseau neuronal contrôlant l'IA

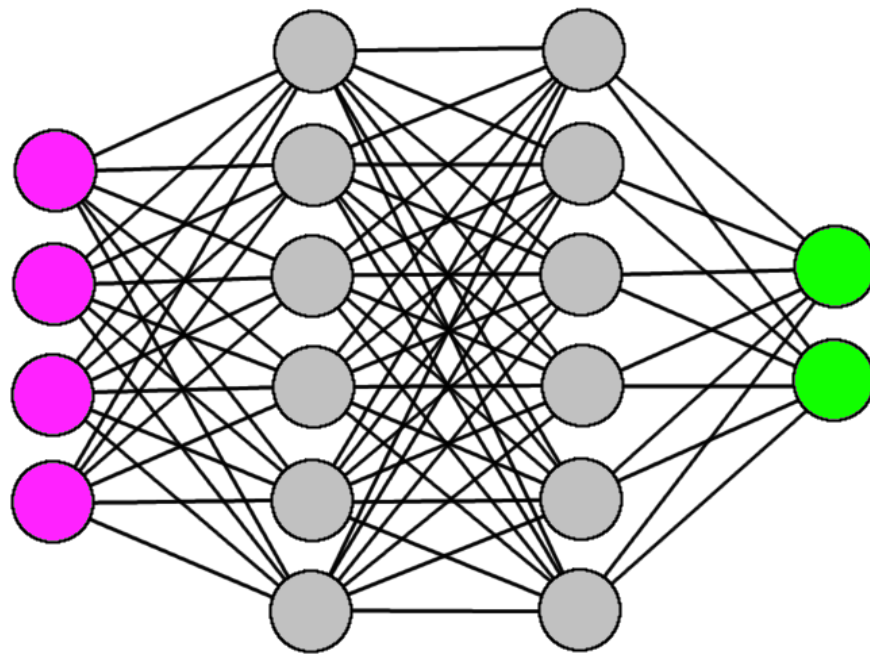
Une fois le jeu codé, il fallait que le T-Rex soit soumis à un réseau neuronal. La librairie synaptic.js a été utilisée. Elle permet de créer un perceptron très simplement, et possède des fonctions qui vont m'être utiles par la suite (notamment les exportations en JSON des perceptrons). Le T-Rex possède donc un Cerveau (Brain.js) qui contient lui-même un perceptron.

Le perceptron a 4 neurones d'entrées qui correspondent respectivement à :

- La vitesse courante du jeu
- La distance séparant le T-Rex du prochain obstacle
- La largeur du prochain obstacle
- Le type du prochain obstacle (CACTUS ou PTERODACTYL)

La valeur d'entrée doit être comprise entre 0 et 1, on effectue donc un ajustement proportionnel des valeurs d'entrée avant de les mettre dans les neurones d'entrée.

A chaque tour dans la boucle de jeu, le réseau neuronal est activé, et, en fonction des inputs, sort deux outputs liés aux actions Sauter et Se Baisser. Si la valeur de ces neurones est supérieure à 0,5 (la valeur varie entre 0 et 1), alors le T-Rex saute ou se baisse selon le neurone sortant.



*Perceptron à 4 couches (18 neurones au total)*

En rose : Neurones d'entrée

En vert : Neurones de sortie

### Squelette de l'algorithme génétique

Par ailleurs, le squelette de l'algorithme génétique est implémenté. Concrètement, il existe des signatures pour les fonctions de :

1. Sélection des meilleurs réseaux neuronaux,
2. Croisement entre les meilleurs réseaux (pour créer de nouveaux individus),
3. Mutation de ces nouveaux individus.

Le calcul de la fitness (ici le nombre de cactus sautés) a lieu au fur et à mesure d'une partie.

## Travail restant

### Implémentation de l'algorithme génétique

À présent, il reste à implémenter les fonctions de l'algorithme génétique.

Appelons 'individu' un perceptron relié à un T-Rex.

Lors de l'initialisation, on crée une population de N génomes (perceptrons aléatoirement instanciés).

On calcule leur fitness lors d'une partie, puis, quand la partie se termine, on sélectionne les meilleurs pour les reproduire (crossing-over puis mutations)

Et on relance une partie avec les nouveaux individus pour calculer leur fitness, et ainsi de suite jusqu'à avoir une IA jouant aussi bien –voire mieux- qu'un humain.

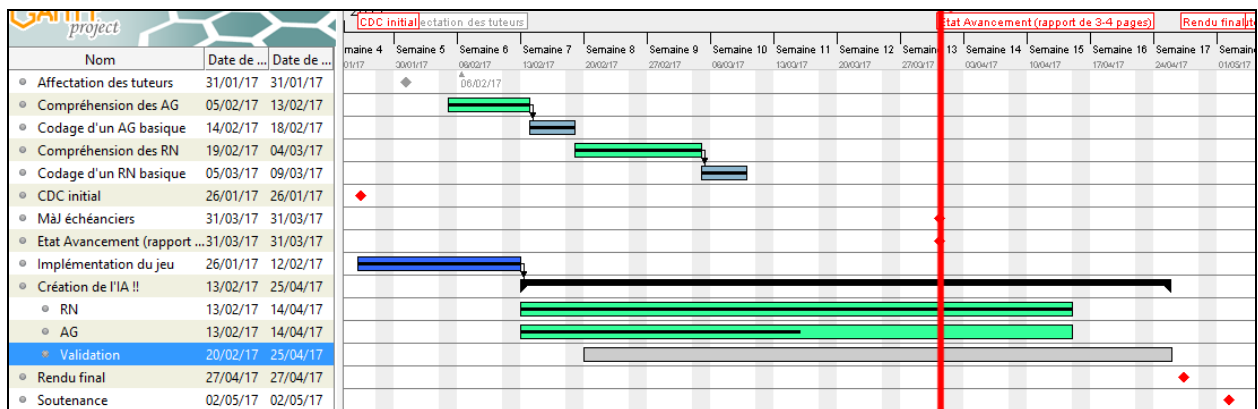
### Affichage des données importantes

Certaines données sont pertinentes à afficher comme par exemple un graphe montrant l'évolution de la fitness moyenne d'une génération autour du temps. Cela permet de confirmer visuellement qu'il y a évolution et permet de détecter les phases de stagnation.

## Conclusion

Au niveau de la gestion de projet, je suis légèrement en retard sur l'implémentation de l'algorithme génétique.

Ci-dessous le Gantt simplifié et à jour. La ligne rouge marque la date d'aujourd'hui.



Gantt du projet – la ligne rouge correspond au 31 mars 2017

Je ne suis pas serein à 100% quant à l'objectif à atteindre. J'espère qu'après implémentation de l'algorithme génétique, le T-Rex apprendra directement. Sinon, le « débogage » va être compliqué.