

# **FOREST, mode d'emploi**

**Jeu d'Othello®**  
**par Olivier Casile**  
[forest\\_othello@orange.fr](mailto:forest_othello@orange.fr)

Version 9.3  
Avril 2024

## **Introduction**

FOREST est un programme de d'Othello® dont les premières versions datent de 1978. A cette époque il était écrit en LSE (*Langage Symbolique d'Enseignement*) et tournait laborieusement sur le CII Mitra® 15 de mon lycée, équipé d'un processeur 16 bits, de 4ko de RAM à tores de ferrite, d'un disque dur de 400 Ko et de disquettes 8". Le nom du programme, FOREST, est dû au grand nombre d'algorithmes arborescents qu'il recèle. Le programme a ensuite beaucoup évolué à la fin des années 1990, à l'âge d'or des programmes d'Othello, pour parvenir à une version 3.7 comportant beaucoup de fonctionnalités qui ont fait le bonheur des joueurs de club. Puis il est entré dans un long sommeil, mes activités professionnelles ayant pris le dessus.

Mais en 2017, année de la résurrection de *l'Intelligence Artificielle*, j'ai pensé qu'Othello serait un excellent exercice de mise en œuvre des techniques de *Deep Learning*, et j'ai repris mon programme afin de rénover son interface utilisateur, d'enlever toutes les fonctionnalités superflues, de moderniser les algorithmes alpha-beta et PVS, et d'affiner le réglage de sa fonction d'évaluation tout en en développant une nouvelle basée sur un réseau de neurones. Ce long travail de rénovation, qui dura 18 mois, donna les versions 4.0 à 4.5 du programme.

La mise au point du réseau de neurone fut extrêmement longue, car il n'y a pas de règle pour obtenir le bon compromis entre précision, complexité, et performance. Le réseau est développé en Python avec Keras et TensorFlow™, puis ré-implementé en C et C++ dans le programme pour la performance. La phase d'apprentissage supervisé du réseau s'effectue sur une bibliothèque de plus d'un million de parties et de 90 millions de positions dont le résultat exact est connu. La version de FOREST que vous avez installé dispose d'un réseau de neurones assez sophistiqué dont la géométrie intègre les symétries du jeu, combiné à une recherche alpha-beta de profondeur 14, qui donne une force de jeu similaire aux meilleurs programmes classiques utilisant une recherche alpha-beta de profondeur 26, ce qui montre bien l'apport du réseau de neurones comme fonction d'évaluation. FOREST, dont la force de jeu était moyenne dans ses versions antérieures, a maintenant rattrapé le niveau des meilleurs programmes comme Zebra, Edax, Ntest, ou le tout récent Egaroucid.

Un autre axe d'amélioration du programme est la bibliothèque d'ouverture, indispensable pour que le programme aborde le milieu de partie dans une position équilibrée et puisse ensuite faire la différence avant la fin de partie qui est déterministe. Jusqu'à Forest 4.3, la bibliothèque d'ouverture était basée sur la base Thor, contenant plus de 100.000 parties jouées par des joueurs humains et des ordinateurs depuis les années 1980. Cette base est très imprécise, car les parties jouées pas les humains et les moins bons programmes sont loin d'être optimales. La bibliothèque d'ouverture est maintenant déterminée plus rigoureusement en faisant jouer le programme contre lui-même, selon un algorithme de Monte-Carlo.

Les principales fonctions offertes par FOREST sont les suivantes :

- 6 niveaux de difficulté, du débutant au joueur confirmé ;
- Choix de l'ouverture ;
- Choix des ouvertures XOT ;
- Large bibliothèque de positions précalculées en ouverture et fin de partie ;
- Conseils de l'ordinateur et affichage de la variante principale ;
- Navigation dans les coups d'une partie ;
- Echange des rôles en cours de partie ;

- Changement du niveau de difficulté en cours de partie ;
- Importation et exportation de parties ;
- Essai de coups et évaluation complète de position ;
- Règles du jeu et stratégies ;
- Version française et anglaise.

## **Compatibilité**

FOREST est un programme **64 bits** nécessite un PC doté d'un processeur Intel ou AMD, de 4 Go de RAM ou plus, et fonctionnant sous les systèmes d'exploitation Microsoft® Windows™ x64 version 10 ou plus récent.

- Depuis la version 4.5, FOREST représente l'othellier sous forme de Bitboards, et utilise l'instruction machine **Popcnt** pour compter les pions.
- Depuis la version 5.2, il utilise de plus le jeu d'instructions vectorielles **AVX2**, et ne peut donc s'exécuter sur les anciens processeurs ne possédant pas ces instructions (processeurs Intel antérieurs à la génération Haswell de 2012, et processeurs AMD antérieurs à la génération Excavator de 2015). FOREST teste le support de ces instructions lors de son lancement, et vous signalera si elles sont absentes sur votre ordinateur.
- Enfin, depuis la version 6.1, FOREST utilise le multi-threading pour accélérer ses calculs et gagner en profondeur d'analyse. Il requiert donc un processeur disposant d'au moins 4 cœurs logiques pour s'exécuter.

Si votre ordinateur ne dispose pas d'un processeur suffisant pour exécuter la dernière version de FOREST, vous avez toujours la possibilité d'utiliser l'une des versions antérieures de FOREST, disponibles sur ma page Web <http://ocasile.pagesperso-orange.fr/forest.htm> .

## **Installation**

FOREST est programme « portable » ne nécessitant pas de programme d'installation. Il est en général fourni sous forme d'un fichier Zip qu'il suffit de décompresser dans un dossier de votre choix. De là, vous pouvez directement exécuter le programme FOREST.exe, ou définir un lien sur ce programme que vous pourrez disposer sur le bureau Windows ou épingler à la barre des tâches. FOREST utilise la base de registre de Windows pour se souvenir de sa position à l'écran et des options que vous avez sélectionné.

Bien que je fasse attention de ne pas introduire de virus ni d'adware dans FOREST, je vous recommande de toujours passer votre anti-virus sur un programme téléchargé avant de l'installer.

## **Mode d'emploi**

Si vous ne connaissez rien aux règles du jeu d'Othello, commencer par lancer FOREST et lisez la *Règle du Jeu* hypertexte accessible par le *menu aide* à droite de la barre de menu. Vous pouvez aussi vous apprendre sur le site de la Fédération Française d'Othello : <http://www.ffothello.org/>

Si, par contre, vous êtes familier avec Othello, vous pouvez lancer FOREST et jouer immédiatement en sélectionnant, à l'aide de la souris, une case licite de l'Othellier. FOREST retourne alors les pions que vous avez entourés, joue à son tour, puis vous rend la main pour le tour suivant. Ceci jusqu'à la fin de la partie, où FOREST vous donnera le score final. Notez que FOREST marque d'une croix les cases licites du jeu. La couleur de cette croix précise la couleur, noire ou blanche, du pion que vous devez poser. Si au cours de la partie un joueur doit passer, FOREST l'indique par une boîte de dialogue apparaissant sur l'écran.

Après les premières parties au niveau amateur, le niveau de difficulté par défaut, vous pourrez commencer à explorer les différentes possibilités de FOREST :

## La barre de menu

- **Partie** permet de lancer une nouvelle partie ou de quitter le programme.
- **Edition** permet de défaire le dernier tour (le dernier coup de l'adversaire et votre précédent coup) afin de rejouer ; de copier la partie dans le clipboard (au format « F5D6... ») ; d'importer une partie depuis le clipboard ; de changer de niveau de jeu ; ou de modifier les options de jeu.
- **Affichage** permet d'ajouter ou d'enlever des éléments d'affichage, comme l'affichage des cases jouables du jeu ou de la valeur des cases jouables dans la bibliothèque de positions précalculées.
- **Information** permet d'afficher la règle du jeu, un résumé du contenu de la bibliothèque de positions précalculées, des informations sur le réseau de neurones évaluant les positions, ou les informations générales sur le programme.

## La barre d'outils propose les boutons suivants :

- Lancement d'une **nouvelle partie** avec choix d'options.
- **Inversion des joueurs**, l'ordinateur joue donc immédiatement puis c'est à votre tour.
- **Symétrie centrale** de l'othellier.
- **Symétrie** de l'othellier par rapport à la **diagonale descendante**.
- **Symétrie** de l'othellier par rapport à la **diagonale montante**.
- **Retour au début** de la partie.
- Retour d'un **coup en arrière** parmi les coups déjà joués.
- **Retour en arrière** d'un **tour complet** (le dernier coup de l'adversaire et votre précédent coup) afin de rejouer.
- Avance d'un **coup en avant** parmi les coups déjà joués.
- **Déplacement** jusqu'au **dernier coups joué**.
- **Evaluation rapide** des cases jouables.

## Les options de nouvelle partie

- Choix du **joueur qui commence**, vous-même ou FOREST.
- **Niveau de jeu** : débutant, débutant +, amateur (choix par défaut), amateur +, expert, et maître.
- **Choix de l'ouverture**, c'est-à-dire de la séquence des premiers coups : quelconque (aléatoire), XOT (tirage aléatoire d'une position de départ équilibrée), ou une ouverture classique désignée par son nom.

## Les options du programme

- **Conseil du meilleur coup** à l'utilisateur, signalé par un cercle rouge autour du centre de la case correspondante.
- L'**anticipation** du coup de l'adversaire, qui permet à FOREST de commencer à calculer son prochain coup pendant que son adversaire réfléchit.
- Affichage des **informations sur les coups** joués par FOREST, comme leur probabilité de gain ou de partie nulle en début de partie, ou les scores prévus en fin de partie.
- **Utilisation**, par FOREST, de sa **bibliothèque d'ouvertures** en début de partie, ce qui améliore son niveau de jeu.

## Les ouvertures standard

Depuis qu'il existe des tournois d'Othello, et tout comme d'autres jeux comme les échecs, les joueurs expérimentés ont appris qu'il existe des ouvertures (la suite des premiers coups) qui sont favorables, et d'autres qui le sont moins. Les ouvertures favorables sont donc plus jouées que les autres, et il existe une nomenclature (plus ou moins officielle) des ouvertures, que vous pourrez trouver ici :

<http://www.ffothello.org/strategie/repertoire-douvertures/>

FOREST propose une quarantaine de ces ouvertures, parmi les plus jouées, dans les options de nouvelle partie. Il suivra la séquence des coups de l'ouverture choisie tant que vous la suivrez vous-même, puis jouera les coups suivant selon son algorithme d'évaluation.

### Les ouvertures XOT

Les bons joueurs ayant tendance à user et abuser des ouvertures favorables qu'ils connaissent, parfois jusqu'au 20-ème coup ou plus, les tournois peuvent devenir monotones. Il a donc été établi, à l'aide de deux excellents programmes d'Othello (Edax et Ntest), une liste d'environ 3600 ouvertures qui au bout de 8 coups donnent une position neutre, c'est-à-dire où ni l'un ni l'autre des joueurs n'a d'avantage positionnel. Cette liste, dite « XOT » peut être consultée à l'adresse suivante :

<http://berg.earthlingz.de/xot/>

FOREST inclue cette liste et tirera aléatoirement l'une de ces ouvertures à votre demande, puis affichera la position résultante à l'écran et vous laissera jouer, avec les pions Noirs.

### La simulation et l'étude de coups

Sur une position donnée, un clic gauche de la souris permet normalement de jouer puis de déclencher automatiquement le jeu de l'ordinateur. Mais un **clic droit de la souris** permet de jouer un coup licite sans déclencher le jeu de l'ordinateur, ce qui permet de rejouer des parties connues, de forcer des coups pour les étudier, ou d'explorer des ouvertures.

### Les raccourcis clavier

- **Ctrl-N** interrompt la partie en cours pour en débiter une nouvelle, en vous permettant de choisir les options de la partie.
- **Ctrl-U** revient en arrière d'un tour complet. Le dernier coup de l'adversaire est annulé ainsi que votre précédent coup, et c'est à vous de rejouer.
- **Ctrl-X** charge immédiatement un début de partie XOT sur l'othellier, et laisse le trait à l'utilisateur. Si vous souhaitez que FOREST commence il suffit d'appuyer sur le bouton d'inversion des joueurs dans la barre d'outils.
- **Ctrl-C** copie la partie en cours dans le clipboard, sous forme de chaîne de caractères (au format « F5D6... »), afin de pouvoir l'importer dans un traitement de texte ou dans un autre programme.
- **Ctrl-V** importe une partie depuis le clipboard. La partie doit être représentée sous forme de chaîne de caractères (au format « F5D6... »). Un contrôle de validité est effectué, et c'est l'utilisateur qui a le trait. Si vous préférez que FOREST commence il suffit d'appuyer sur le bouton d'inversion des joueurs dans la barre d'outils.
- **Ctrl-S** produit dans un fichier une trace de la partie jusqu'au dernier coup joué.

### Le fichier de trace

A la fin de chaque partie, FOREST sauvegarde dans son dossier d'exécution un fichier de trace de nom « Forest <date> <heure>.log ». Ce fichier contient :

- Le numéro de version de FOREST.
- La date et l'heure de fin de la partie.
- Le nom Windows de l'ordinateur et du compte utilisateur.
- Le nombre de cœurs logiques du processeur, et le nombre de cœurs effectivement utilisés par le programme.
- La vitesse de calcul de FOREST en positions analysées par secondes.
- La taille mémoire de la table de transpositions, un cache des coups évalués durant les recherches itératives.
- Les caractéristiques de la bibliothèque de positions précalculées.
- Le niveau de jeu de la partie.
- Le pion joué par FOREST, Noir (X) ou Blanc (O).

- La liste des coups joués, avec des informations sur leur valeur calculée par FOREST, et leur variante principale.
- L'othellier final avec le nombre de pions Noirs et Blancs.
- La chaîne de caractères représentant la partie.

Les fichiers de trace sont très utiles à l'amélioration de FOREST, et permettent à l'auteur du programme de lui faire apprendre de ses succès et de ses erreurs. Donc si vous en collectez quelques-uns, en particulier ceux où vous avez battu FOREST au niveau maître, l'auteur du programme (moi) vous sera reconnaissant de les lui envoyer par mail à l'adresse [forest\\_othello@orange.fr](mailto:forest_othello@orange.fr).

## **Notes d'implémentation**

FOREST est entièrement écrit en C++ 2017, et compilé à l'aide du compilateur Microsoft® Visual Studio 2019. FOREST est composé d'une vingtaine de modules, et comporte plus de 10.000 lignes de code.

FOREST est capable d'évaluer plus de 800.000 positions par seconde avec un processeur Intel Core i9 11900. Il utilise des algorithmes de type alpha-beta PVS avec table de transposition, et une fonction d'évaluation basée sur un réseau de neurones travaillant sur des patterns de bords/coins et sur la mobilité des joueurs. Sa profondeur d'analyse est de 14 coups au maximum en milieu de partie et de 28 coups maximum en fin de partie.

Les niveaux de Jeu sont définis comme suit :

- Débutant : jeu du coup immédiatement déterminé par le « Policy Network » de FOREST, donc avec une profondeur d'analyse nulle ;
- Débutant + : jeu du coup minimisant la valeur du prochain coup de l'adversaire calculée par le « Value Network » de FOREST, ce qui correspond à un Minimax de profondeur 1 ;
- Amateur : bibliothèque d'ouverture limitée à 16 coups, profondeur d'analyse de 7 coups en milieu de partie, recherche du gain dans les 18 derniers coups, et maximisation du score exact dans les 16 derniers coups ;
- Amateur + : bibliothèque d'ouverture limitée à 18 coups, profondeur d'analyse de 9 coups en milieu de partie, recherche du gain dans les 22 derniers coups, et maximisation du score exact dans les 20 derniers coups ;
- Expert : bibliothèque d'ouverture limitée à 20 coups, profondeur d'analyse de 12 coups en milieu de partie, recherche du gain dans les 24 derniers coups, et maximisation du score exact dans les 22 derniers coups ;
- Maître : bibliothèque d'ouverture illimitée, profondeur d'analyse de 14 coups en milieu de partie, recherche du gain dans les 28 derniers coups, et maximisation du score exact dans les 26 derniers coups ;

La bibliothèque d'ouvertures et de finales contient environ 2 millions de positions provenant de plus d'un million de parties de haut niveau jouées par ordinateur. Pour ceux qui souhaitent utiliser ces positions pour entraîner leur propre programme, voici la structure du fichier FOREST.bib les contenant :

Champ	Type et Taille	Contenu
Entête		
CRC32	uint32_t	0 pour l'instant
N_POSITIONS	int32_t	Nombre total de positions dans la bibliothèque
TAILLE	int32_t	Taille totale des données contenues dans le fichier
PADDING	uint32_t	Réservé pour usage futur
DATE	__time64_t	Date de génération de la bibliothèque : durée en secondes écoulées depuis le 1er janvier 1970 à minuit

Tables des nombres de positions		
TABLE_TOTAL	uint32_t [60]	Nombres Totaux de positions, pour les coups joués de 0 à 60 (donc de 60 à 0 cases vides)
TABLE_EXACT	uint32_t [60]	Nombres de positions résolues exactement, pour les coups joués de 0 à 60 (donc de 60 à 0 cases vides)
Tables des Min et Max de Valeurs de position		
TABLE_MIN	float [60]	Table des Minimums des Valeurs des positions, pour les coups joués de 0 à 60
TABLE_MAX	float [60]	Table des Maximums des Valeurs des positions, pour les coups joués de 0 à 60
Listes des Bitboards des positions de la base, classées par nombre de coups joués (de 0 à 60), puis par bitboard croissant.		
LISTE_BB_N	uint64_t [N_POSITIONS]	Liste des bitboards du joueur Noir pour l'ensemble des positions. Un bit $2^N$ à 1 correspondant à un pion noir dans la case N, avec N=0 pour la case A1
LISTE_BB_B	uint64_t [N_POSITIONS]	Liste des bitboards du joueur Blanc pour l'ensemble des positions.
Table des statistiques des positions, avec pour chaque position:		
PROB_GAIN	float	Probabilité de Gain de la position, selon l'ensemble des parties déjà jouées depuis cette position
PROB_NULLE	float	Probabilité de partie Nulle de la position
VAL_REL	int8_t	Valeur Relative de la position pour le Joueur, entre -64 et +64
CONFIANCE	uint8_t	Confiance en la valeur de la position. 100 indique que la valeur est exacte, une valeur inférieure indique que la valeur est estimée (par minimax de toutes les parties issues de la position)

Notes:

- **Le format de la bibliothèque de positions change à partir de la version 9.1 de FOREST**, afin d'améliorer les performances de la recherche de position.
- Les nombres sont stockés au format « little indian », donc pour les processeurs x86.
- Les positions sont exprimées pour le joueur ayant le trait (le premier bitboard d'une position est donc pour noir ou blanc selon le joueur qui a le trait).
- Les bitboards d'une position sont Uniques, c'est-à-dire indépendants des 8 symétries possibles du jeu. En pratique il s'agit du minimum des 8 couples de bitboards des 8 symétries.

## **Nouveautés**

Améliorations par rapport à la version 9.2:

- Parallélisation des recherches Alpha-Beta avec l'algorithme ABDADA inventé par Jean-Christophe Weill ;
- Meilleur traitement des cases X définitives par le réseau de neurones.

Améliorations prévues dans de futures versions :

- Refonte de la Table de Transposition pour de meilleures performances ;
- Amélioration du multi-threading ;



- Approfondissement de la bibliothèque de positions précalculées.

## **Mentions légales**

- *Othello* est une marque déposée de Kabushiki Kaisha MegaHouse
- *Reversi* est une marque déposée de Ravensburger AG
- *Windows* et *Visual Studio* sont des marques déposées de Microsoft Corporation
- *TensorFlow* est une marque de Google Inc.

## **Licence d'utilisation**

Le programme FOREST a été développé par **Olivier Casile**. FOREST peut être diffusée librement, mais ne peut en aucun cas faire l'objet d'un usage commercial.

Un grand merci à:

- **Christophe Lanuit**, pour ses conseils et les discussions techniques sur les algorithmes de jeu, sa fonction de hashing super rapide, les bases de position qui m'ont permis d'améliorer le niveau de jeu de FOREST, et son élégant générateur de coups basé sur les instructions PEXT/PDEP.
- **Matthias Berg** pour son accord à l'inclusion de la liste des ouvertures XOT dans FOREST (<http://xot.xmav.eu>).
- **Richard Delorme**, pour ses conseils sur la parallélisation des algorithmes alpha-beta, et pour avoir publié et mis à jour le code source de son programme EDAX (<https://github.com/abulmo/edax-reversi>), qui sert de sparring partner à FOREST.
- **Tom Kerrigan**, pour sa version simplifiée de l'algorithme ABDADA (inventé par **Jean-Christophe Weill**, <https://www.chessprogramming.org/ABDADA>), qui a largement inspiré l'implémentation du « Lazy SMP » dans FOREST: [http://www.tckerrigan.com/Chess/Parallel\\_Search/Simplified\\_ABDADA/](http://www.tckerrigan.com/Chess/Parallel_Search/Simplified_ABDADA/).
- **Emmanuel Bougeard**, pour ses tests comparatifs de FOREST, et pour ses études approfondies sur les ouvertures.
- **Gunnar Andersson et Lars Ivansson**, dont l'excellent programme Zebra sert aussi de sparring partner à FOREST.
- **Chris Welty**, pour son extraordinaire bibliothèque d'ouverture (<http://www.orbanova.com/nbook/>), qui me sert à valider celle de FOREST.
- **Emmanuel Lazard**, pour l'autorisation de transcrire son manuel d'initiation « A la découverte d'Othello » sous forme d'aide hypertexte.
- **Bruno de la Boisserie** pour ses conseils en stratégie, pour la diffusion de FOREST à ses débuts, et pour son indispensable « *Anthologie des programmes d'Othello* ».

Je serai très heureux de recevoir remarques, suggestions, bugs, parties gagnées à l'adresse: [forest\\_othello@orange.fr](mailto:forest_othello@orange.fr).

St Laurent du Var, le 29 avril 2024