



Ceci est un extrait électronique d'une publication de
Diamond Editions :

<http://www.ed-diamond.com>

Retrouvez sur le site tous les anciens numéros en vente par
correspondance ainsi que les tarifs d'abonnement.

Pour vous tenir au courant de l'actualité du magazine, visitez :

<http://www.gnulinuxmag.com>

Ainsi que :

<http://www.linux-pratique.com>

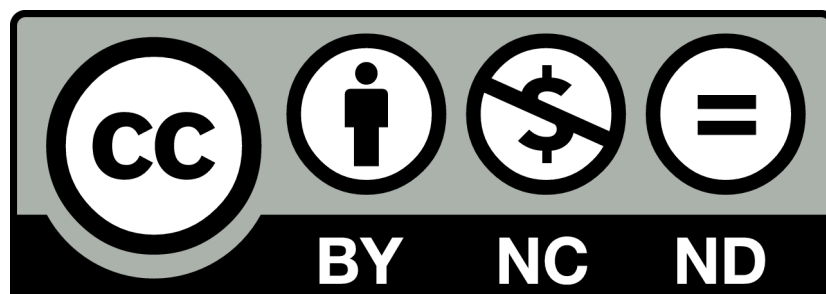
et

<http://www.miscmag.com>



Ceci est un extrait électronique d'une publication de Diamond Editions

<http://www.ed-diamond.com>



Creative Commons

Paternité - Pas d'Utilisation Commerciale - Pas de Modification 2.0 France

Vous êtes libres :

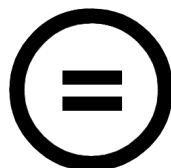
- de reproduire, distribuer et communiquer cette création au public.



Paternité. Vous devez citer le nom de l'auteur original de la manière indiquée par l'auteur de l'oeuvre ou le titulaire des droits qui vous confère cette autorisation (mais pas d'une manière qui suggérerait qu'ils vous soutiennent ou approuvent votre utilisation de l'oeuvre).



Pas d'Utilisation Commerciale. Vous n'avez pas le droit d'utiliser cette création à des fins commerciales.



Pas de Modification. Vous n'avez pas le droit de modifier, de transformer ou d'adapter cette création.

A chaque réutilisation ou distribution de cette création, vous devez faire apparaître clairement au public les conditions contractuelles de sa mise à disposition.

- Chacune de ces conditions peut être levée si vous obtenez l'autorisation du titulaire des droits.
- Rien dans ce contrat ne diminue ou ne restreint le droit moral de l'auteur ou des auteurs.

Ceci est le Résumé Explicatif du Code Juridique. La version intégrale du contrat est attachée en fin de document et disponible sur :

<http://creativecommons.org/licenses/by-nc-nd/2.0/fr/legalcode>

→ Concepts fondamentaux des algorithmes évolutionnistes

Jean-Baptiste Mouret

EN DEUX MOTS Les algorithmes évolutionnistes sont des méthodes d'optimisation globale grossièrement inspirées par la théorie néo-darwinienne. Ce premier article présente les concepts fondamentaux et les grandes variantes. Il sera suivi d'un deuxième les appliquant à la mise au point automatique de la morphologie et du « cerveau » de créatures capables de se déplacer.

De l'origine des espèces

En publiant *L'Origine des espèces*, en 1859, Charles Darwin expose pour la première fois sa théorie de l'évolution, désormais communément admise. Celle-ci repose sur les observations suivantes :

- Il existe au sein de chaque espèce de nombreuses variations, chaque individu étant différent ;
- Les ressources naturelles étant finies, il naît rapidement plus d'êtres vivants que la nature ne peut en nourrir ; il en résulte une lutte pour l'existence (*struggle for life*) entre chaque organisme ;
- Les individus survivants possèdent des caractéristiques qui les rendent plus aptes à survivre ; Darwin baptise ce concept « sélection naturelle » ;
- Les organismes survivants transmettent leurs avantages à leur descendance ; l'accumulation au cours des générations des petites différences entre chaque branche généalogique crée de nouvelles espèces de plus en plus aptes à survivre.

Ces quatre principes de la théorie darwinienne ont été étendus dans les années 1930 pour incorporer les découvertes en génétique. Cette théorie synthétique de l'évolution a été baptisée néo-darwinisme et correspond aux bases de l'approche en vigueur dans la communauté scientifique.

Dans le langage génétique moderne, les molécules d'ADN, constituées d'une suite de nucléotides, sont le support du génotype – l'ensemble des gènes – dont l'expression physiologique est dénommée le phénotype. Il n'existe pas de définition précise de la notion de « gène » qui soit universellement reconnue. Sans entrer dans les détails, on peut considérer qu'un gène est une séquence de nucléotides codant pour un caractère particulier, le plus souvent une protéine.

L'échelle de temps sur laquelle se déroule le processus évolutionniste naturel est si grande qu'elle se prête mal à l'expérimentation.

Mais, fort de cette compréhension de la question de l'origine des espèces, comment résister au désir de nous incarner en démiurge ?

Réaliser ce fantasme sur un ordinateur afin de créer des machines plus « intelligentes » anime Turing dès 1948. Mais il faut attendre les travaux autour des stratégies d'évolution au milieu des années 1960, puis ceux de John H. Holland sur les algorithmes génétiques, peu de temps après, pour observer les premières utilisations pratiques appliquant les concepts du néo-darwinisme à l'informatique.

Ces recherches ont mené à quatre grandes familles d'algorithmes évolutionnistes : les algorithmes génétiques, les stratégies d'évolution, la programmation génétique et la programmation évolutionniste.

Par souci de complétude, on peut signaler deux autres catégories d'algorithmes évolutionnistes, que nous n'aborderons pas dans cet article : les algorithmes à estimation de distribution (EDA pour *Estimation of Distribution Algorithms*) et les algorithmes de co-évolution).

Les quatre grandes familles d'algorithmes évolutionnistes

Ces techniques reposent toutes sur les mêmes principes de base. Tout d'abord, elles travaillent sur une population, initialement aléatoire, d'individus représentant chacun une solution au problème traité.

Chaque solution est représentée par un génotype et s'exprime sous la forme d'un phénotype. Il est ensuite nécessaire de mettre au point une fonction d'adaptation (*fitness*) qui permet de distinguer les phénotypes les plus performants.

Ces derniers ont une probabilité plus importante de léguer à leur descendance leur génotype. Les règles régissant cette transmission de gènes sont décrites sous la forme d'opérateurs génétiques. Trois opérateurs sont requis :

- L'opérateur de sélection qui décrit la manière dont les candidats sont choisis pour la reproduction ;
- L'opérateur de croisement (*cross-over*, parfois aussi appelé « opérateur de recombinaison ») qui correspond à la méthode utilisée pour mélanger les génotypes des parents ;
- L'opérateur de mutation qui décrit la fréquence et la nature des changements affectant le génotype lors de la transmission des gènes.

Dans le cas des algorithmes génétiques, les solutions potentielles sont représentées par des chaînes binaires (par exemple 0001). L'opérateur principal est le croisement. La mutation est utilisée pour ajouter de la diversité à la population.

Dans le cadre de la programmation génétique, une solution est encodée sous la forme d'un programme. Le programme est

exécuté afin de permettre le calcul de la fitness. Le génotype est vu comme un arbre où chaque nœud représente une sous-fonction, une structure de contrôle, un opérateur ou un terminal.

L'opérateur principal est le croisement, celui-ci consistant à des échanges de sous-arbres. À l'inverse, la programmation évolutionniste et les stratégies d'évolution utilisent essentiellement l'opérateur de mutation. Le premier cas correspond à l'évolution de programmes, le second à celle d'une liste de paramètres.

Actuellement, la distinction entre ces quatre approches est de plus en plus floue, le génotype étant souvent constitué d'un mélange de structures complexes (arbres, graphes, listes de paramètres,...).

La différence entre ces quatre catégories est donc essentiellement d'ordre historique. En conséquence, on utilisera dans la suite de cet article le terme d'« algorithmes évolutionnistes », qui regroupe les différentes familles d'algorithmes sus-citées.

Afin de clarifier la description, on peut décrire en pseudocode le processus général, commun à toutes les approches évolutionnistes :

- 1. Génération d'une population aléatoire ;
- 2. Évaluation de la fitness de chaque individu ;
- 3. Sélection des individus pour la reproduction (opérateur de sélection) ;
- 4. Reproduction des individus sélectionnés à l'aide de l'opérateur de croisement ; on obtient un ensemble de nouveaux individus ;
- 5. Application de l'opérateur de mutation sur les nouveaux individus ;
- 6. Constitution de la nouvelle génération en se basant sur les nouveaux individus ;
- 7. Retour à l'étape 2.

Nous reviendrons par la suite sur l'implémentation de chacune de ces étapes.



Fig. 1 : Le réseau de neurone contrôlant ce dirigeable lenticulaire (Projet Alpha, en coopération avec l'AnimatLab) a été créé par une méthode évolutionniste.



Fig. 2 : Créature dont la morphologie et le contrôleur ont été obtenus à l'aide d'une méthode évolutionniste (Projet GOLEM, par Lipson et Pollack, université de Brandeis). Les parties du corps et les circuits de contrôle sont d'abord évolués en simulation, puis une technologie de prototypage rapide est utilisée pour obtenir un robot réel.

Quelques exemples d'application

Les méthodes évolutionnistes ont été exploitées avec succès dans des applications très variées. On peut ainsi relever de nombreuses utilisations dans le monde industriel, par exemple pour concevoir des armatures métalliques, pour le choix de certains paramètres lors de la conception des avions ou encore pour la mise au point d'une turbine.

Ce type de problème nécessite typiquement l'exploration d'un espace de plus d'une centaine de variables, sous quelques dizaines de contraintes.

Cependant, les réalisations les plus spectaculaires sont probablement situées dans le domaine de la vie artificielle. Les méthodes évolutionnistes ont ainsi été utilisées, par exemple, pour construire des réseaux de neurones capables de contrôler des robots hexapodes. Alors que les premières générations parcourent difficilement quelques centimètres avant de rencontrer lourdement le sol, leurs arrière-arrière-... petits enfants, quelques centaines de générations plus tard, se déplacent fort honorablement.

Des méthodes proches ont été exploitées pour créer le réseau de neurone contrôlant le dirigeable de la *figure 1*. Peut-être plus impressionnant encore, Lipson et Pollack, deux chercheurs américains, ont utilisé une machine à prototypage « rapide » et des méthodes évolutionnistes pour définir, puis fabriquer, la morphologie et les contrôleurs de créatures telles que celle de la *figure 2*. Les chercheurs ont retrouvé dans les mouvements effectués par leurs machines de nombreuses solutions utilisées dans la nature.

Des métaheuristiques d'optimisation globale performantes

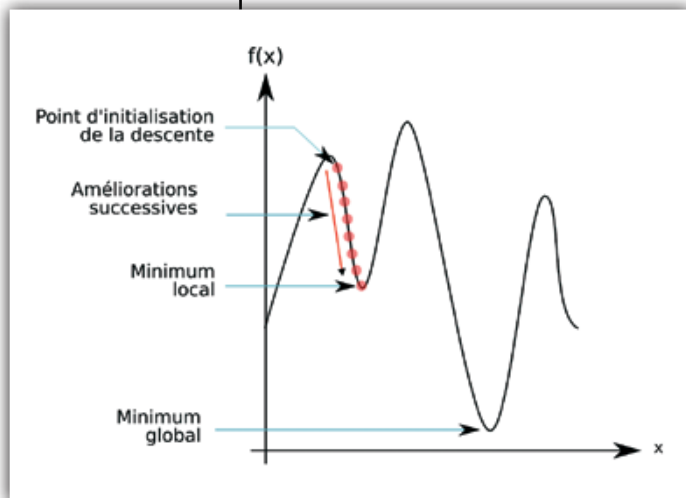


Fig. 3 : Illustration de l'optimisation d'une fonction $y=f(x)$ à l'aide d'une descente de gradient : il est facile de ne trouver qu'un minimum local.

D'un point de vue plus mathématique, on classe les algorithmes évolutionnistes parmi les métaheuristiques d'optimisation globale aux côtés du recuit simulé et de la recherche tabou.

Sous ces consonances barbares se dissimule d'abord la notion, fondamentale en informatique, d'« optimisation ». Une quantité très importante des problèmes posés en intelligence artificielle et en recherche opérationnelle s'expriment en effet naturellement sous la forme d'une fonction dont on cherche un ou des optimums.

La recherche de ces derniers est appelée « optimisation ». Si la fonction à optimiser est linéaire ou dérivable, le domaine de la programmation mathématique fournit de nombreuses méthodes envisageables.

La descente de gradient constitue certainement la méthode la plus courante (figure 3). Mais, à moins que la fonction soit strictement convexe, le minimum trouvé à l'aide de cette méthode ne sera généralement qu'un extremum local : il existe le plus souvent d'autres minima plus petits.

Dans le cadre des algorithmes évolutionnistes, on cherche à maximiser (ce qui revient à minimiser l'opposé) la fonction de fitness.

Travailler avec un ensemble de solutions en parallèle (figure 4) permet d'éviter le piège des minimums locaux (bien que ces méthodes ne trouvent pas toujours le minimum global).

De plus, le domaine des fonctions optimisables avec les algorithmes évolutionnistes est particulièrement vaste comparé aux autres méthodes d'optimisation. Puisqu'elle n'est utilisée qu'à l'étape 2, la seule contrainte explicite concernant la fonction de fitness est d'être évaluable pour tous les individus.

En particulier, il n'est pas nécessaire que cette fonction soit dérivable. Toutefois, les méthodes évolutionnistes reposent implicitement sur une hypothèse de quasi-continuité de la fonction fitness. Leur utilisation suppose, en effet, que des solutions différant par des changements élémentaires soient associées à des performances voisines.

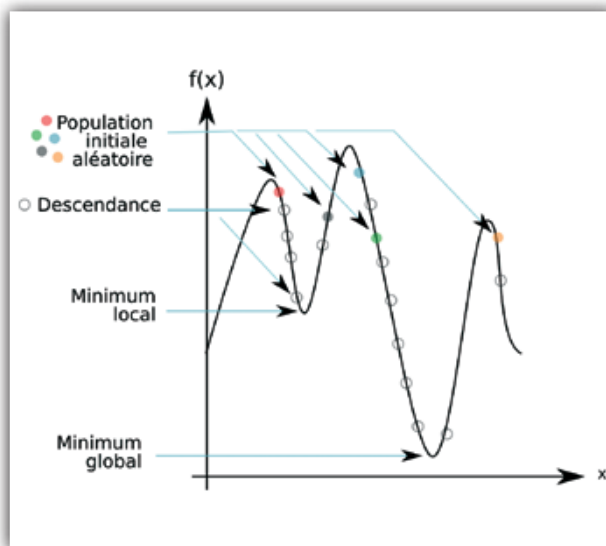


Fig. 4 : Illustration de l'optimisation d'une fonction $y=f(x)$ à l'aide d'un algorithme évolutionniste : utiliser une population augmente la probabilité de trouver un minimum global.

Utiliser de telles méthodes ne requiert donc aucune compréhension mathématique détaillée de la fonction à optimiser. Celle-ci peut notamment prendre la forme d'un algorithme, d'une simulation physique d'un robot ou même de tests effectués sur des robots réels. Cette caractéristique fait écrire à John H. Holland qu'elles « peuvent résoudre des problèmes que même leurs créateurs ne comprennent pas complètement ».

Le programmeur ne décrit que les outils pour manipuler les solutions et non pas leur fonctionnement. Il est alors courant d'observer des solutions performantes tout en étant incapable d'expliquer leurs rouages internes. Citons quelques exemples de fonction de fitness utilisées :

- Pour un problème type « voyageur de commerce », la longueur du chemin correspondant à chaque génotype ;
- Pour l'apprentissage dans un perceptron multi-couche, l'erreur quadratique moyenne sur la base d'apprentissage ;
- Dans le cadre de l'évolution d'un contrôleur pour un robot marcheur, la distance parcourue par chaque individu pendant une période de temps fixée.

La possibilité d'optimiser des fonctions sans en connaître le fonctionnement interne est à l'origine de la partie « méta » (du grec ancien « après ») du terme métaheuristique : il s'agit d'un processus général, de haut niveau, une sorte de stratégie

que l'on spécialise au problème traité par la spécification des opérateurs génétiques.

On classe ce processus dans la catégorie des heuristiques, car le résultat optimal n'est pas garanti.

Un autre avantage par rapport aux méthodes d'optimisation classiques réside dans le parallélisme implicite des algorithmes évolutionnistes. Puisqu'il faut évaluer la performance de chaque individu indépendamment, rien ne s'oppose à répartir cette évaluation entre plusieurs machines ou plusieurs microprocesseurs. Il est ainsi possible d'exploiter efficacement la puissance de calcul d'un ensemble de machines.

Codage

Le choix du codage, c'est-à-dire la manière dont sont représentées les solutions potentielles manipulées, est certainement le facteur le plus déterminant pour la performance des algorithmes évolutionnistes. Ce choix étant intrinsèquement lié à celui des opérateurs, on sélectionnera toujours un couple codage-opérateurs. Plusieurs questions sont soulevées lorsque l'on aborde le problème du choix du codage :

- Le codage utilisé couvre-t-il toute solution envisageable avec la même probabilité ?
- Est-il possible de mettre au point un opérateur de croisement de telle sorte que les enfants aient de bonnes chances de combiner les avantages des parents ?
- Comment définir un opérateur de mutation permettant une « bonne » exploration de l'espace de recherche ?

Il n'existe malheureusement pas de réponse générale à ces questions.

Pour des raisons historiques et théoriques, le codage le plus souvent étudié est le codage binaire, où chaque solution est encodée sous la forme d'une chaîne de bits de longueur fixe.

Sachant que c'est ainsi que nos ordinateurs fonctionnent, il est aisé de comprendre que cette représentation permet de coder sous une forme unifiée n'importe quel objet, du nombre entier non signé au graphe complexe. Comme nous le verrons par la suite, la définition des opérateurs de croisement et de mutation est, de plus, particulièrement facile.

Imaginons, par exemple, que l'on souhaite optimiser la taille d'un parallélogramme rectangle défini par sa longueur, sa hauteur et sa largeur. Pour simplifier l'exemple, supposons que ces trois quantités soient des nombres entiers compris entre 0 et 15. On peut alors décrire chacune d'elles en utilisant un nombre binaire de 4 bits. Un exemple de solution potentielle peut avoir pour génotype 0001 0111 01010. Le phénotype correspondant serait un parallélogramme de 1 de long, 7 de haut et 10 de large.

Deux individus aux phénotypes très proches, par exemple les entiers 7 et 8 ont cependant un codage très différent, en l'occurrence 0111 et 1000. Il est donc difficile de passer du premier individu vers le second via les opérateurs de mutation et de croisement alors que la solution 1000 pourrait être associée à une fitness plus importante. Le codage de Gray résout ce problème en s'assurant que deux nombres « adjacents » ne diffèrent pas de plus d'un bit. Voici les procédures pour transformer un nombre codé en binaire vers le codage de

Gray et réciproquement :

Binaire $B=b_{1...m}$ vers codage de Gray

$G=g_{1...m}$:

$$g_1 = b_1$$

pour k de 2 à m :

$$g_k = b_{k-1} \text{ XOR } b_k$$

Codage de Gray $G=g_{1...m}$ vers binaire

$B=b_{1...m}$:

$$v = g_1$$

pour k de 2 à m :

$$\text{si } g_k = 1 \text{ alors } v = \text{NOT } v$$

$$b_k = v$$

D'autres approches, parfois plus intuitives et plus performantes, sont possibles. On utilise ainsi souvent des génotypes constitués simplement de vecteurs de réels ou d'entiers. Si l'on reprend l'exemple précédent du parallélogramme, on peut ainsi choisir comme génotype un vecteur de trois entiers.

Nous reviendrons dans la section consacrée aux opérateurs génétiques sur l'intérêt d'une telle approche.

Enfin, des structures plus évoluées tels des arbres ou des graphes peuvent être utilisées comme génotypes à partir du moment où les opérateurs peuvent être implémentés.

Opérateurs de sélection

La manière la plus classique d'implémenter l'opérateur de sélection est la méthode de la roulette biaisée. Ce principe se base sur l'image d'une roulette de casino telle que la probabilité de sélectionner un individu particulier soit proportionnelle à la valeur de sa fonction d'adaptation. Ainsi, les meilleurs individus auront une probabilité plus importante d'être sélectionnés pour la reproduction.

Cependant, des tests empiriques ont montré qu'il était plus efficace de se baser sur le classement des individus dans la population, appelé leur rang, plutôt que sur la valeur de la fitness.

Dans ce dernier cas, si un individu obtient une fitness nettement meilleure que les autres, il risque d'envahir rapidement la population. Cette réduction rapide de la diversité génétique mène facilement à une convergence prématurée vers un unique minimum local.

Pour sélectionner les individus en fonction de leur rang, on commence par les classer en fonction de leur fitness par ordre décroissant puis on définit la probabilité de tirer un rang i comme le multiple de la probabilité

Des tests empiriques ont montré qu'il était plus efficace de se baser sur le classement des individus dans la population, appelé leur rang, plutôt que sur la valeur de la fitness.

de tirer le rang du dessous par un facteur multiplicatif. On aboutit à la formule suivante, permettant de tirer un index i :

$$i = n - \frac{n}{\log(\kappa + 1)} \cdot \log(\kappa \cdot \text{rand}(1) + 1)$$

où $\kappa = c^{n+1} - 1$, n est le nombre d'individus que l'on souhaite prendre en compte, c un coefficient par exemple égal à 1.1 et $\text{rand}(1)$ un générateur aléatoire renvoyant un nombre réel compris entre 0 et 1. Une autre méthode classique de sélection est celle du tournoi. On tire successivement des couples d'individus et on sélectionne celui qui possède la meilleure fitness. On itère le processus jusqu'à ce que l'on ait sélectionné suffisamment de parents.

Ces deux approches peuvent être combinées au principe de l'élitisme. Afin d'éviter la perte des meilleurs individus, on peut choisir de les recopier directement vers la génération suivante. On évite ainsi que la meilleure fitness puisse décroître et que les très bonnes solutions soient éliminées par la nature stochastique (aléatoire) de l'opérateur de sélection. On conserve généralement 5 à 10 % des meilleurs individus.

Comme ce sera le cas pour les autres opérateurs, il existe très peu de base théorique pour guider le choix de la méthode de sélection. Seuls des tests concernant votre application spécifique vous permettront de déterminer l'approche la plus efficace.

Opérateurs de croisement

Le croisement permet d'améliorer les performances de la population en manipulant la structure des génotypes des individus. Idéalement, les caractéristiques des enfants devraient être le fruit d'un mélange de celles des parents, ce mélange pouvant se révéler bénéfique comme négatif du point de vue de la fitness.

Le croisement associé au codage binaire et au codage de Gray est appelé le croisement par découpage (*slicing crossover*). Le génotype de chaque parent étant défini par une chaîne de bits, un nombre, appelé point de croisement, est tiré aléatoirement. Il correspond à la position du bit de la chaîne où celle-ci va être coupée en deux morceaux (*figure 5*).

On intervertit ensuite les sous-chaînes terminales des deux parents pour obtenir le génotype de deux enfants. La procédure peut être étendue à un croisement à plusieurs points.

Notons que si le génotype est constitué de plusieurs gènes, par exemple s'il code plusieurs variables, ce type de croisement peut couper au milieu d'un gène.

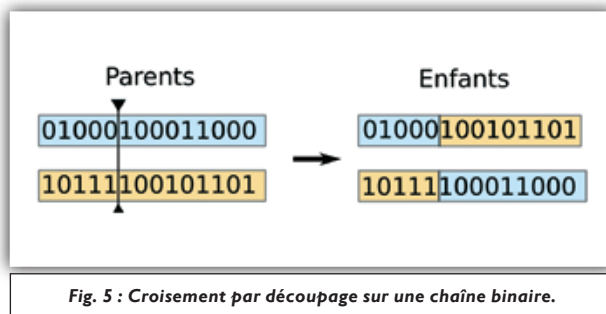


Fig. 5 : Croisement par découpage sur une chaîne binaire.

L'application du codage binaire – et donc du croisement précédemment décrit – à des problèmes utilisant des nombres réels est souvent décevante.

De nombreuses stratégies sont envisageables pour coder des nombres réels en binaire, à commencer par la représentation IEEE 754 des nombres flottants utilisée dans nos ordinateurs.

Conserver une différence bits à bits faible pour deux valeurs proches avec ces représentations est difficile. Mais, surtout, le résultat du croisement de deux nombres réels codés de cette manière est assez contre-intuitif.

Ainsi, le réel 2.5625 est représenté, selon la norme IEEE sur 32 bits, par la chaîne binaire 0 10000000 010010000000000000000000 et le nombre 0.36 par la chaîne 0 011111010 1110000101000111101100.

Essayons de les croiser. En découpant au niveau du troisième bit (en partant de la gauche), on obtient pour l'un des enfants la chaîne 0 0110000 000100100000000000000000 qui correspond approximativement à $5.97 \cdot 10^{-10}$.

Il est difficile de saisir le rapport entre l'enfant et les parents ! Avant de tenter d'investiguer d'autres codages, on peut se demander s'il n'est pas plus pratique d'utiliser comme génotype un vecteur de nombre réel sans s'intéresser à sa représentation binaire. Après tout, il suffit d'être capable de définir les opérateurs de croisement et de mutation adaptés.

À quoi peut-on s'attendre lorsque l'on croise 2.5625 et 0.36 ? La réponse la plus intuitive est certainement la moyenne des deux réels, car elle semble combiner les « avantages » des deux parents.

Cependant, l'utilisation de ce type de croisement comporte un fort risque d'obtenir rapidement une population très homogène dont tous les individus seraient des copies d'un individu « moyen ».

Une méthode souvent préconisée, le croisement barycentrique, s'inspire de ce concept de moyenne et y ajoute une notion de hasard inspirée du point de croisement utilisé dans le croisement binaire.

Pour générer deux enfants e_1 et e_2 à partir des parents p_1 et p_2 , on commence par tirer un nombre α au hasard dans l'intervalle $[-0.5; 1.5]$. On applique ensuite les formules :

$$e_1 = \alpha p_1 + (1 - \alpha) p_2$$

$$e_2 = \alpha p_2 + (1 - \alpha) p_1$$

Une méthode souvent préconisée, le croisement barycentrique, s'inspire de ce concept de moyenne et y ajoute une notion de hasard

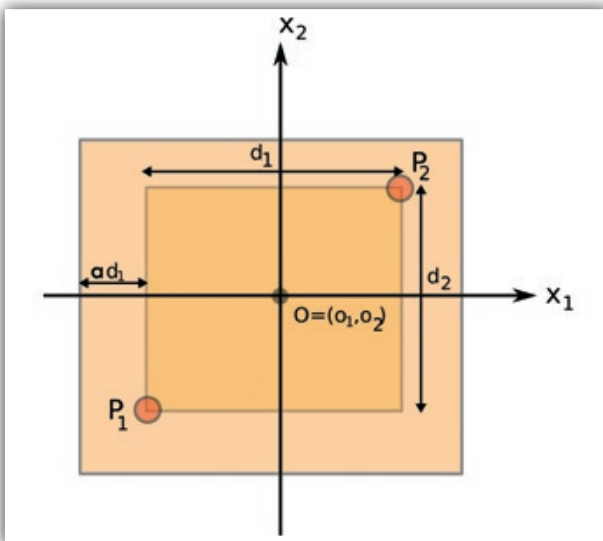


Fig. 6 : Croisement $BLX-\alpha$ pour des vecteurs de deux réels. Un point est choisi selon une loi uniforme dans le grand rectangle.

Le croisement $BLX-\alpha$ est une autre méthode souvent utilisée et donnant de bons résultats dans un grand nombre de situations. Supposons que chaque individu soit décrit par un couple de nombres réels (la méthode se généralise sans difficulté à des vecteurs de taille quelconque). On peut placer les deux parents sur un plan (figure 6) ; ils définissent alors les deux coins d'un rectangle. La valeur α permet de fixer un facteur d'échelle à ce rectangle. Le croisement $BLX-\alpha$ choisit un point aléatoire selon une loi uniforme dans ce rectangle centré en $O=(o_1, o_2)$, c'est-à-dire dans l'intervalle $[o_1 - \frac{d_1}{2} - \alpha d_1; o_1 + \frac{d_1}{2} + \alpha d_1]$ sur l'axe x_1 et dans l'intervalle $[o_2 - \frac{d_2}{2} - \alpha d_2; o_2 + \frac{d_2}{2} + \alpha d_2]$ sur l'axe x_2 . De nombreuses valeurs pour α peuvent être utilisées, mais la plus souvent choisie est $\alpha=0,5$.

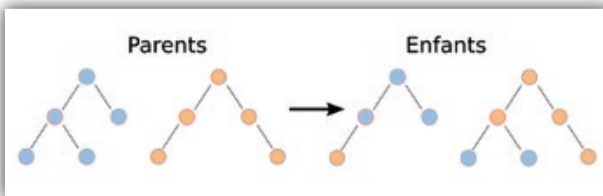


Fig. 7 : Croisement de deux arbres : échange de sous-arbres

Pour terminer ce rapide tour d'horizon des méthodes de croisement, signalons la méthode de croisement classiquement utilisée en programmation génétique et, par extension, dans tous les problèmes où les solutions sont représentées par des arbres. Dans cette situation, on sélectionne aléatoirement un nœud dans chaque parent, par exemple le deuxième nœud lors d'un parcours en profondeur et on échange les sous-arbres correspondants (figure 7).

Opérateurs de mutation

L'opérateur de mutation apporte aux processus évolutionnistes la possibilité d'explorer l'ensemble de l'espace de recherche. Comme c'est le cas dans les stratégies d'évolution, il est tout à fait possible de ne pas utiliser l'opérateur de croisement et de s'appuyer uniquement sur l'opérateur de mutation.

La mutation opérant sur le génotype, elle dépend donc du codage utilisé. Cependant, le concept sous-jacent est le même quel que soit le codage : ajouter des petites modifications au génotype afin d'explorer de nouvelles solutions potentielles au problème traité.

La définition de la mutation dans le cadre du codage binaire est particulièrement simple.

Dans un premier temps, une ou plusieurs positions sont tirées. Le ou les bits correspondants sont alors remplacés par une valeur tirée aléatoirement (en l'occurrence 0 ou 1).

Comme dans le cas du croisement, cette approche fonctionne mal sur des nombres réels. La stratégie la plus courante est d'ajouter du bruit à certains membres du vecteur, tirés aléatoirement.

Différentes distributions de probabilité, centrées sur la valeur du gène avant la mutation, peuvent être envisagées.

Dans le cas d'une loi uniforme, une variation est choisie uniformément dans un intervalle fixé autour de la valeur actuelle du gène. Une distribution gaussienne peut aussi être utilisée.

La largeur de l'intervalle choisi dans le cas de la loi uniforme et le paramètre d'écart type dans celui de la loi gaussienne peuvent être délicats à choisir. Si celui-ci est trop faible, l'espace de recherche risque ne pas être suffisamment exploré.

À l'inverse, s'il est trop important, l'algorithme peut avoir des difficultés à affiner les solutions alors qu'il est proche de l'optimum. Une approche intéressante pour résoudre ce problème est de coder ces paramètres dans le génotype.

Ainsi, les solutions explorant rapidement l'espace seront favorisées au début de l'algorithme et auront naturellement tendance à s'éteindre au fur et à mesure que le processus converge vers l'optimum.

Améliorations classiques

Il existe de nombreuses variantes des algorithmes évolutionnistes. La simple combinaison des différents opérateurs présentés amène à un nombre important de possibilités.

Nous n'avons cependant décrit que les méthodes les plus classiques et la littérature fournit de nombreux autres codages et opérateurs. De plus, beaucoup d'améliorations, tentant de résoudre certains problèmes observés, ont été proposées.

Le croisement $BLX-\alpha$ est une autre méthode souvent utilisée et donnant de bons résultats dans un grand nombre de situations.

La mise à l'échelle (*scaling*) de la fitness est un outil souvent mis en place lorsque la valeur de la fitness est utilisée explicitement dans le processus de sélection.

Afin d'éviter que des individus trop forts ne dominent rapidement la population, au risque de prématurément converger vers un optimum local, il est possible de réduire artificiellement l'écart entre les individus en transformant la fitness. À l'opposé, on constate parfois que les bons individus ont du mal à se distinguer et il peut être utile d'augmenter la différence de fitness entre eux.

Ces opérations peuvent être aisément réalisées en appliquant des fonctions classiques à la valeur de la fitness. Par exemple, prendre le carré de la fitness augmentera les différences.

Plus généralement, on définit la mise à l'échelle exponentielle d'une fitness f pour obtenir une nouvelle fitness f_s à la génération n de la façon suivante : $f_s = f^{k(n)}$. Pour k proche de zéro, les écarts de fitness sont réduits, pour k nul, la mise à l'échelle ne change pas la fitness et pour k supérieur à 1, les écarts sont augmentés.

Il peut être intéressant de faire varier k en fonction de la génération, essentiellement en l'augmentant durant le processus. Ainsi, au début de l'algorithme, on préserve le maximum de diversité puis, lorsque l'on s'approche de la fin, on se concentre sur les individus les plus performants.

Le partage (*sharing*) est une autre amélioration souvent mise en œuvre pour pousser l'algorithme à explorer l'ensemble de l'espace de recherche. Le principe utilisé consiste à diminuer la fitness des solutions similaires.

Pour cela, il faut d'abord définir une distance – par exemple la distance euclidienne – entre chaque individu. On choisit ensuite un voisinage, par exemple une sphère de rayon 2 autour de chaque solution.

On peut alors compter le nombre d'individus présents dans ce voisinage en pondérant par la distance (plus un individu est éloigné, moins on le prend en compte) ; on obtient une valeur baptisée « coefficient de niche ».

Enfin, la fitness « brute » de chaque individu est divisée par le coefficient de niche. Cette méthode donne de bons résultats. Néanmoins, elle requiert le calcul de la distance de chaque individu à tous les autres.

Si la population contient N individus, cette approche nécessite, par conséquent, N^2 calculs supplémentaires à chaque génération.

Cette complexité peut très souvent être négligée car, dans de nombreuses applications, le calcul de la fitness se révèle être beaucoup plus long que tous les autres calculs.

Dans les situations où l'espace de recherche est très grand, il est parfois très difficile de générer aléatoirement des individus intéressants alors qu'on n'en tire qu'une centaine.

Dans ce cas, on observe souvent une variabilité importante des résultats : deux exécutions successives de l'algorithme évolutionniste peuvent mener à des résultats radicalement différents.

Ce problème a souvent ses racines au niveau de la dépendance de l'algorithme à la population initiale. En effet, si la population de départ ne peut pas mener à des solutions efficaces par améliorations successives simples, les algorithmes évolutionnistes rencontrent généralement de grandes difficultés à trouver de bons optimums.

Une approche simple mais donnant souvent de bons résultats pour éviter ce problème consiste à tirer avant la première génération un très grand nombre d'individus aléatoires (par exemple 50 fois la population de travail).

Après évaluation de la fitness pour chacun d'eux, on ne conserve que les meilleurs. Ceux-ci constituent la population initiale de l'algorithme évolutionniste. Ainsi, en explorant suffisamment l'espace de départ, on réduit le risque de ne trouver aucune solution prometteuse dans la population initiale.

Gestion de contraintes

Dans la plupart des problèmes réels, certaines solutions doivent être évitées même si elles pourraient obtenir une bonne fitness.

Par exemple, dans le cas de l'optimisation d'un contrôleur pour un robot, on peut vouloir éviter les solutions susceptibles de casser le matériel.

La méthode des pénalités est une approche classique pour intégrer des contraintes dans le cadre des algorithmes évolutionnistes.

Elle repose sur un désir simple : un individu violant les contraintes ne doit pas pouvoir être mieux classé qu'un individu les respectant.

Pour cela, il suffit de soustraire une pénalité suffisamment grande à la fitness des individus ne respectant pas les contraintes.

Par exemple, si toutes les fitness sont comprises entre 0 et 1, il suffit de soustraire 10 à la fitness de tous les individus violant les contraintes pour obtenir l'effet souhaité. Si plusieurs contraintes sont présentes, il est possible de sommer les pénalités afin de privilégier les individus n'en transgressant qu'une partie.

Multicritères

Jusqu'à présent, nous n'avons envisagé que le cas où l'on pouvait aisément réduire la performance d'une solution pour le problème posé à une unique valeur numérique que l'on a appelée la fitness.

En pratique, très peu de problèmes peuvent être exprimés de cette façon, car ils mettent en jeu un compromis entre plusieurs objectifs potentiellement incompatibles.

Un individu violant les contraintes ne doit pas pouvoir être mieux classé qu'un individu les respectant.

Imaginons, par exemple, quelqu'un souhaitant acheter une voiture et pour lequel deux critères sont importants, le prix et le confort. Ces deux critères ne sont pas compatibles, une voiture plus confortable étant plus chère qu'une moins confortable. Comment trouver le meilleur compromis ?

La technique la plus souvent utilisée est d'agréger les différents objectifs en un seul à l'aide d'une somme pondérée : pour optimiser n objectifs $O_1(x), O_2(x), \dots, O_n(x)$, on cherche le minimum de la fonction $f(x) = \alpha_1 O_1(x) + \alpha_2 O_2(x) + \dots + \alpha_n O_n(x)$ où $\alpha_1, \alpha_2, \dots, \alpha_n$ sont des réels représentant le poids de chaque critère. Ces réels fixent un compromis particulier entre les différents critères. Le choix des poids est problématique, car généralement il n'existe pas de méthode théorique pour les estimer.

L'utilisateur doit alors recourir à une approche par essais/erreurs qui peut prendre beaucoup de temps, car le processus d'optimisation doit être relancé pour chaque test.

De plus, on peut montrer que cette approche ne permet pas de trouver tous les compromis.

Des variantes des algorithmes évolutionnistes apportent de nombreuses solutions élégantes à ces problèmes d'optimisation multicritères. Leur description nécessiterait malheureusement un article complet et nous ne pourrions, par conséquent, pas en dire plus.

Les curieux peuvent toutefois nourrir leur moteur de recherche préféré des mots-clefs « *multiobjective optimization* » et consulter la page <http://www.lania.mx/~ccoello/EMOO/EMOObib.html>. Un bon algorithme est celui proposé par C. Fonseca baptisé « MOGA », dont l'implémentation se révèle assez aisée.



LIENS

Tout d'abord, deux livres de vulgarisation à propos du néo-darwinisme. Bien que les algorithmes présentés dans cet article soient plutôt éloignés des théories biologiques, celles-ci restent une source d'inspiration intéressante.

► Pierre-Henry Gouyon, Jean-Pierre Henry et Jacques Arnould, *Les avatars du gène*, Belin (1997). Un livre assez « scolaire », mais qui présente efficacement les différentes théories actuelles.

► Richard Dawkins, *Le gène égoïste*, Éditions Odile Jacob (1990). Dawkins décrit dans ce livre une vision de l'évolution centrée sur le gène et non sur l'individu. Bien qu'encore débattue, cette approche semble être de plus en plus admise dans la communauté scientifique.

Par ailleurs, on ne peut éviter de citer les livres de D.E. Goldberg et de M. Mitchell qui contribuèrent beaucoup à l'utilisation des algorithmes génétiques et restent des références :

► David E. Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning*, Addison Wesley (1989).

► Melanie Mitchell, *An Introduction to Genetic Algorithms*, MIT Press (1995).

Conclusion

Les algorithmes évolutionnistes constituent des outils d'optimisation efficaces pouvant s'appliquer à de nombreux problèmes.

Ils permettent notamment de traiter des problèmes où la fonction objectif ne présente pas de propriétés mathématiques exploitables (dérivabilité, convexité,...) et s'implémentent facilement sur des architectures parallèles.

Néanmoins, leur aspect magique et leur potentiel ne doivent pas masquer de nombreuses limitations :

► Les outils théoriques disponibles pour effectuer les choix optimaux (sélection du codage et des opérateurs génétiques) sont quasi inexistantes. Au final, il est nécessaire d'expérimenter de nombreuses combinaisons pour obtenir de bons résultats et il est souvent difficile de justifier les choix effectués.

► La généralité des algorithmes évolutionnistes se paie le plus souvent par des performances inférieures à celles obtenues à l'aide d'un algorithme spécifique, exploitant les propriétés mathématiques des fonctions à optimiser.

► De nombreux paramètres, par exemple les taux de mutation, sont difficiles à régler et nécessitent de nombreuses expérimentations.

► Certains problèmes se prêtent mal à la définition d'opérateurs génétiques efficaces.

Si les solutions trouvées sont généralement bonnes, il est rare que les méthodes évolutionnistes trouvent le minimum global en dehors des problèmes « jouets ».

Cette critique peut cependant s'étendre à l'ensemble des métaheuristiques d'optimisation (recuit simulé, recherche tabou,...) : la recherche, dans le cas général, du minimum d'une fonction est un problème particulièrement difficile.

Il n'existe, à ce jour, aucun algorithme efficace capable de trouver cet optimum de manière certaine.

Jean-Baptiste Mouret,

Creative Commons

Paternité - Pas d'Utilisation Commerciale - Pas de Modification 2.0

Creative Commons n'est pas un cabinet d'avocats et ne fournit pas de services de conseil juridique. La distribution de la présente version de ce contrat ne crée aucune relation juridique entre les parties au contrat présenté ci-après et Creative Commons. Creative Commons fournit cette offre de contrat-type en l'état, à seule fin d'information. Creative Commons ne saurait être tenu responsable des éventuels préjudices résultant du contenu ou de l'utilisation de ce contrat.

Contrat

L'Oeuvre (telle que définie ci-dessous) est mise à disposition selon les termes du présent contrat appelé Contrat Public Creative Commons (dénommé ici « CPCC » ou « Contrat »). L'Oeuvre est protégée par le droit de la propriété littéraire et artistique (droit d'auteur, droits voisins, droits des producteurs de bases de données) ou toute autre loi applicable. Toute utilisation de l'Oeuvre autrement qu'explicitement autorisée selon ce Contrat ou le droit applicable est interdite.

L'exercice sur l'Oeuvre de tout droit proposé par le présent contrat vaut acceptation de celui-ci. Selon les termes et les obligations du présent contrat, la partie Offrante propose à la partie Acceptante l'exercice de certains droits présentés ci-après, et l'Acceptant en approuve les termes et conditions d'utilisation.

1. Définitions

- a. « **Oeuvre** » : oeuvre de l'esprit protégeable par le droit de la propriété littéraire et artistique ou toute loi applicable et qui est mise à disposition selon les termes du présent Contrat.
- b. « **Oeuvre dite Collective** » : une oeuvre dans laquelle l'oeuvre, dans sa forme intégrale et non modifiée, est assemblée en un ensemble collectif avec d'autres contributions qui constituent en elles-mêmes des oeuvres séparées et indépendantes. Constituent notamment des Oeuvres dites Collectives les publications périodiques, les anthologies ou les encyclopédies. Aux termes de la présente autorisation, une oeuvre qui constitue une Oeuvre dite Collective ne sera pas considérée comme une Oeuvre dite Dérivée (telle que définie ci-après).
- c. « **Oeuvre dite Dérivée** » : une oeuvre créée soit à partir de l'Oeuvre seule, soit à partir de l'Oeuvre et d'autres oeuvres préexistantes. Constituent notamment des Oeuvres dites Dérivées les traductions, les arrangements musicaux, les adaptations théâtrales, littéraires ou cinématographiques, les enregistrements sonores, les reproductions par un art ou un procédé quelconque, les résumés, ou toute autre forme sous laquelle l'Oeuvre puisse être remaniée, modifiée, transformée ou adaptée, à l'exception d'une oeuvre qui constitue une Oeuvre dite Collective. Une Oeuvre dite Collective ne sera pas considérée comme une Oeuvre dite Dérivée aux termes du présent Contrat. Dans le cas où l'Oeuvre serait une composition musicale ou un enregistrement sonore, la synchronisation de l'oeuvre avec une image animée sera considérée comme une Oeuvre dite Dérivée pour les propos de ce Contrat.
- d. « **Auteur original** » : la ou les personnes physiques qui ont créé l'Oeuvre.
- e. « **Offrant** » : la ou les personne(s) physique(s) ou morale(s) qui proposent la mise à disposition de l'Oeuvre selon les termes du présent Contrat.
- f. « **Acceptant** » : la personne physique ou morale qui accepte le présent contrat et exerce des droits sans en avoir violé les termes au préalable ou qui a reçu l'autorisation expresse de l'Offrant d'exercer des droits dans le cadre du présent contrat malgré une précédente violation de ce contrat.

2. Exceptions aux droits exclusifs. Aucune disposition de ce contrat n'a pour intention de réduire, limiter ou restreindre les prérogatives issues des exceptions aux droits, de l'épuisement des droits ou d'autres limitations aux droits exclusifs des ayants droit selon le droit de la propriété littéraire et artistique ou les autres lois applicables.

3. Autorisation. Soumis aux termes et conditions définis dans cette autorisation, et ceci pendant toute la durée de protection de l'Oeuvre par le droit de la propriété littéraire et artistique ou le droit applicable, l'Offrant accorde à l'Acceptant l'autorisation mondiale d'exercer à titre gratuit et non exclusif les droits suivants :

- a. reproduire l'Oeuvre, incorporer l'Oeuvre dans une ou plusieurs Oeuvres dites Collectives et reproduire l'Oeuvre telle qu'incorporée dans lesdites Oeuvres dites Collectives;
- b. distribuer des exemplaires ou enregistrements, présenter, représenter ou communiquer l'Oeuvre au public par tout procédé technique, y compris incorporée dans des Oeuvres Collectives;
- c. lorsque l'Oeuvre est une base de données, extraire et réutiliser des parties substantielles de l'Oeuvre.

Les droits mentionnés ci-dessus peuvent être exercés sur tous les supports, médias, procédés techniques et formats. Les droits ci-dessus incluent le droit d'effectuer les modifications nécessaires techniquement à l'exercice des droits dans d'autres formats et procédés techniques. L'exercice de tous les droits qui ne sont pas expressément autorisés par l'Offrant ou dont il n'aurait pas la gestion demeure réservé, notamment les mécanismes de gestion collective obligatoire applicables décrits à l'article 4(d).

4. Restrictions. L'autorisation accordée par l'article 3 est expressément assujettie et limitée par le respect des restrictions suivantes :

- a. L'Acceptant peut reproduire, distribuer, représenter ou communiquer au public l'Oeuvre y compris par voie numérique uniquement selon les termes de ce Contrat. L'Acceptant doit inclure une copie ou l'adresse Internet (Identifiant Uniforme de Ressource) du présent Contrat à toute reproduction ou enregistrement de l'Oeuvre que l'Acceptant distribue, représente ou communique au public y compris par voie numérique. L'Acceptant ne peut pas offrir ou imposer de conditions d'utilisation de l'Oeuvre qui altèrent ou restreignent les termes du présent Contrat ou l'exercice des droits qui y sont accordés au bénéficiaire. L'Acceptant ne peut pas céder de droits sur l'Oeuvre. L'Acceptant doit conserver intactes toutes les informations qui renvoient à ce Contrat et à l'exonération de responsabilité. L'Acceptant ne peut pas reproduire, distribuer, représenter ou communiquer au public l'Oeuvre, y compris par voie numérique, en utilisant une mesure technique de contrôle d'accès ou de contrôle d'utilisation qui serait contradictoire avec les termes de cet Accord contractuel. Les mentions ci-dessus s'appliquent à l'Oeuvre telle qu'incorporée dans une Oeuvre dite Collective, mais, en dehors de l'Oeuvre en elle-même, ne soumettent pas l'Oeuvre dite Collective, aux termes du présent Contrat. Si l'Acceptant crée une Oeuvre dite Collective, à la demande de tout Offrant, il devra, dans la mesure du possible, retirer de l'Oeuvre dite Collective toute référence au dit Offrant, comme demandé. Si l'Acceptant crée une Oeuvre dite Collective, à la demande de tout Auteur, il devra, dans la mesure du possible, retirer de l'Oeuvre dite Collective toute référence au dit Auteur, comme demandé.

- b. L'Acceptant ne peut exercer aucun des droits conférés par l'article 3 avec l'intention ou l'objectif d'obtenir un profit commercial ou une compensation financière personnelle. L'échange de l'Oeuvre avec d'autres Oeuvres protégées par le droit de la propriété littéraire et artistique par le partage électronique de fichiers, ou par tout autre moyen, n'est pas considéré comme un échange avec l'intention ou l'objectif d'un profit commercial ou d'une compensation financière personnelle, dans la mesure où aucun paiement ou compensation financière n'intervient en relation avec l'échange d'Oeuvres protégées.
- c. Si l'Acceptant reproduit, distribue, représente ou communique l'Oeuvre au public, y compris par voie numérique, il doit conserver intactes toutes les informations sur le régime des droits et en attribuer la paternité à l'Auteur Original, de manière raisonnable au regard du médium ou au moyen utilisé. Il doit communiquer le nom de l'Auteur Original ou son éventuel pseudonyme s'il est indiqué ; le titre de l'Oeuvre Originale s'il est indiqué ; dans la mesure du possible, l'adresse Internet ou l'Identifiant Uniforme de Ressource (URI), s'il existe, spécifié par l'Offrant comme associé à l'Oeuvre, à moins que cette adresse ne renvoie pas aux informations légales (paternité et conditions d'utilisation de l'Oeuvre). Ces obligations d'attribution de paternité doivent être exécutées de manière raisonnable. Cependant, dans le cas d'une Oeuvre dite Collective, ces informations doivent, au minimum, apparaître à la place et de manière aussi visible que celles à laquelle apparaissent les informations de même nature.
- d. Dans le cas où une utilisation de l'Oeuvre serait soumise à un régime légal de gestion collective obligatoire, l'Offrant se réserve le droit exclusif de collecter ces redevances par l'intermédiaire de la société de perception et de répartition des droits compétente. Sont notamment concernés la radiodiffusion et la communication dans un lieu public de phonogrammes publiés à des fins de commerce, certains cas de retransmission par câble et satellite, la copie privée d'Oeuvres fixées sur phonogrammes ou vidéogrammes, la reproduction par reprographie.

5. Garantie et exonération de responsabilité

- a. En mettant l'Oeuvre à la disposition du public selon les termes de ce Contrat, l'Offrant déclare de bonne foi qu'à sa connaissance et dans les limites d'une enquête raisonnable :
 - i. L'Offrant a obtenu tous les droits sur l'Oeuvre nécessaires pour pouvoir autoriser l'exercice des droits accordés par le présent Contrat, et permettre la jouissance paisible et l'exercice licite de ces droits, ceci sans que l'Acceptant n'ait aucune obligation de verser de rémunération ou tout autre paiement ou droits, dans la limite des mécanismes de gestion collective obligatoire applicables décrits à l'article 4(e);
- b. L'Oeuvre n'est constitutive ni d'une violation des droits de tiers, notamment du droit de la propriété littéraire et artistique, du droit des marques, du droit de l'information, du droit civil ou de tout autre droit, ni de diffamation, de violation de la vie privée ou de tout autre préjudice délictuel à l'égard de toute tierce partie.
- c. A l'exception des situations expressément mentionnées dans le présent Contrat ou dans un autre accord écrit, ou exigées par la loi applicable, l'Oeuvre est mise à disposition en l'état sans garantie d'aucune sorte, qu'elle soit expresse ou tacite, y compris à l'égard du contenu ou de l'exactitude de l'Oeuvre.

6. Limitation de responsabilité. A l'exception des garanties d'ordre public imposées par la loi applicable et des réparations imposées par le régime de la responsabilité vis-à-vis d'un tiers en raison de la violation des garanties prévues par l'article 5 du présent contrat, l'Offrant ne sera en aucun cas tenu responsable vis-à-vis de l'Acceptant, sur la base d'aucune théorie légale ni en raison d'aucun préjudice direct, indirect, matériel ou moral, résultant de l'exécution du présent Contrat ou de l'utilisation de l'Oeuvre, y compris dans l'hypothèse où l'Offrant avait connaissance de la possible existence d'un tel préjudice.

7. Résiliation

- a. Tout manquement aux termes du contrat par l'Acceptant entraîne la résiliation automatique du Contrat et la fin des droits qui en découlent. Cependant, le contrat conserve ses effets envers les personnes physiques ou morales qui ont reçu de la part de l'Acceptant, en exécution du présent contrat, la mise à disposition d'Oeuvres dites Dérivées, ou d'Oeuvres dites Collectives, ceci tant qu'elles respectent pleinement leurs obligations. Les sections 1, 2, 5, 6 et 7 du contrat continuent à s'appliquer après la résiliation de celui-ci.
- b. Dans les limites indiquées ci-dessus, le présent Contrat s'applique pendant toute la durée de protection de l'Oeuvre selon le droit applicable. Néanmoins, l'Offrant se réserve à tout moment le droit d'exploiter l'Oeuvre sous des conditions contractuelles différentes, ou d'en cesser la diffusion; cependant, le recours à cette option ne doit pas conduire à retirer les effets du présent Contrat (ou de tout contrat qui a été ou doit être accordé selon les termes de ce Contrat), et ce Contrat continuera à s'appliquer dans tous ses effets jusqu'à ce que sa résiliation intervienne dans les conditions décrites ci-dessus.

8. Divers

- a. A chaque reproduction ou communication au public par voie numérique de l'Oeuvre ou d'une Oeuvre dite Collective par l'Acceptant, l'Offrant propose au bénéficiaire une offre de mise à disposition de l'Oeuvre dans des termes et conditions identiques à ceux accordés à la partie Acceptante dans le présent Contrat.
- b. La nullité ou l'inapplicabilité d'une quelconque disposition de ce Contrat au regard de la loi applicable n'affecte pas celle des autres dispositions qui resteront pleinement valides et applicables. Sans action additionnelle par les parties à cet accord, lesdites dispositions devront être interprétées dans la mesure minimum nécessaire à leur validité et leur applicabilité.
- c. Aucune limite, renonciation ou modification des termes ou dispositions du présent Contrat ne pourra être acceptée sans le consentement écrit et signé de la partie compétente.
- d. Ce Contrat constitue le seul accord entre les parties à propos de l'Oeuvre mise ici à disposition. Il n'existe aucun élément annexe, accord supplémentaire ou mandat portant sur cette Oeuvre en dehors des éléments mentionnés ici. L'Offrant ne sera tenu par aucune disposition supplémentaire qui pourrait apparaître dans une quelconque communication en provenance de l'Acceptant. Ce Contrat ne peut être modifié sans l'accord mutuel écrit de l'Offrant et de l'Acceptant.
- e. Le droit applicable est le droit français.

Creative Commons n'est pas partie à ce Contrat et n'offre aucune forme de garantie relative à l'Oeuvre. Creative Commons décline toute responsabilité à l'égard de l'Acceptant ou de toute autre partie, quel que soit le fondement légal de cette responsabilité et quel que soit le préjudice subi, direct, indirect, matériel ou moral, qui surviendrait en rapport avec le présent Contrat. Cependant, si Creative Commons s'est expressément identifié comme Offrant pour mettre une Oeuvre à disposition selon les termes de ce Contrat, Creative Commons jouira de tous les droits et obligations d'un Offrant.

A l'exception des fins limitées à informer le public que l'Oeuvre est mise à disposition sous CPCC, aucune des parties n'utilisera la marque « Creative Commons » ou toute autre indication ou logo afférent sans le consentement préalable écrit de Creative Commons. Toute utilisation autorisée devra être effectuée en conformité avec les lignes directrices de Creative Commons à jour au moment de l'utilisation, telles qu'elles sont disponibles sur son site Internet ou sur simple demande.

Creative Commons peut être contacté à <http://creativecommons.org/>.