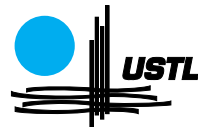

Modèles et simulations informatiques des problèmes de coopération entre agents

Bruno Beaufils

LIFL - Axe CIM - Équipe SMAC



Plan

1. Motivations
2. Dilemme itéré du prisonnier
3. Simulations
4. Conclusions

Motivations

L'équipe SMAC a, entre autre, pour centre d'intérêts :

- les systèmes multi-agents, incluant des agents informatiques ou humains
- les comportements d'agents favorisant la coopération dans le groupe
- l'évolution de la coopération dans des populations d'agents

L'objectif de la thèse a donc été :

- d'étudier en profondeur un modèle formel de coopération entre agents :
 - par la mise en place de simulations
 - par la collecte d'informations sur ce modèle
- d'essayer de déterminer des critères de qualité pour les comportements de façon à favoriser la coopération

Le modèle

Le modèle choisi est le modèle du «*Dilemme itéré du prisonnier*» :

- Modèle de la théorie des jeux introduit à la RAND Corp. (1952)
- Popularisé au début des années 80 par AXELROD
- Très souvent **utilisé** (200 publications entre 1988 et 1994) notamment en économie, psychologie et en biologie théorique
- Étudié surtout dans des espaces continus et non déterministes
- Le modèle simple, souvent considéré comme trop théorique, a souvent été délaissé pour des modèles plus *proches* de la réalité.

⇒ Nous avons :

- étudié le modèle simple trop rapidement mis de côté à notre avis
- adapté celui-ci au cas discret de façon à pouvoir faire des simulations informatiques

Dilemme (itéré) du prisonnier

- Au sens de la théorie des jeux : un jeu simultané, symétrique, à deux joueurs, à somme non nulle, non coopératif où chaque joueur doit choisir une parmi deux cartes : la *Coopération* (C) et la *Trahison* (D)
- Il y a dilemme car l'*intérêt individuel* (\equiv la rationalité au sens économique) (D,D) rapporte moins que l'*intérêt collectif* (C,C)
- Version itérée :
 - les joueurs se rencontrent un certain nombre de fois
 - les joueurs ne connaissent pas le terme du jeu
 - le score des joueurs est la somme des scores de chaque rencontre

Dilemme... $S < P < R < T$

... itéré $S + T < 2R$

	C	D
C	$R = 3, R = 3$	$S = 0, T = 5$
D	$T = 5, S = 0$	$P = 1, P = 1$

T = Tentation

R = Récompense

P = Punition

S = Salaire de la duperie

Stratégies

- À chaque étape un joueur sait ce que son adversaire a joué dans les coups précédents
- Il est alors possible de définir un **comportement** prédéfini pour le jeu : une **stratégie**
- 2 types de stratégies au sens de la théorie des jeux :
 - *pures* en n'utilisant que des choix déterministes (en fonction de l'histoire du jeu)
 - *mixtes* en utilisant des choix non déterministes (distribution de probabilités sur les stratégies pures)
- Quelques exemples de stratégies :
 - all_c
 - all_d
 - per_ccd
 - soft_majo
 - tit_for_tat
 - spiteful

⇒ Identification d'un agent (un joueur) à sa stratégie

Exemples (stratégies)

	1	2	3	4	5	6	7	8	9	10		
all_c	3	3	0	3	3	0	3	3	0	3	=	21
	C	C	C	C	C	C	C	C	C	C		
per_ccd	C	C	D	C	C	D	C	C	D	C	=	36
	3	3	5	3	3	5	3	3	5	3		

tit_for_tat	0	1	1	1	1	1	1	1	1	1	=	9
	C	D	D	D	D	D	D	D	D	D		
all_d	D	D	D	D	D	D	D	D	D	D	=	14
	5	1	1	1	1	1	1	1	1	1		

Évaluation des stratégies

Il existe deux types de méthodes d'évaluation :

1. Le tournoi

- rencontre 2 à 2 de toutes les stratégies d'un panel
- matchs de même longueur inconnue des joueurs
- chaque stratégie est classée en fonction de son score : $\sum V(i|j)$

2. Les évolutions écologiques

- population polymorphe d'individus : plusieurs représentants d'une stratégie d'un panel donné
- facteur d'adaptation d'un individu = somme des scores obtenus face à tous les autres individus
- reproduction proportionnelle des sous-populations de stratégies jusqu'à stabilisation

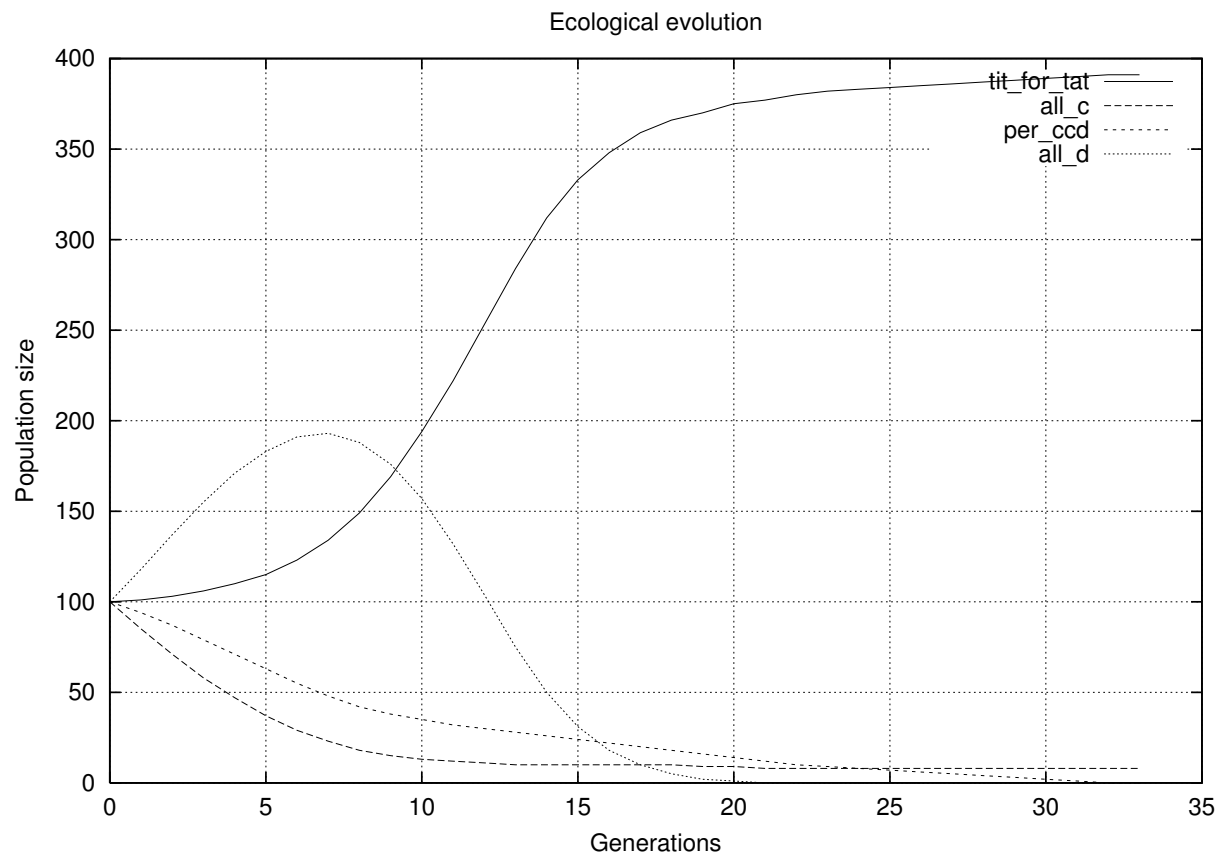
Exemples (tournoi)

	all_c	all_d	per_ccd	tit_for_tat	
all_c	30	0	21	30	81
all_d	50	10	38	14	112
per_ccd	36	3	24	27	90
tit_for_tat	30	9	27	30	96

Classement {
1 all_d
2 tit_for_tat
3 per_ccd
4 all_c

Exemples (évolution)

	1	2	3	4	...
all_c	100	85	71	58	...
all_d	100	118	137	155	...
per_ccd	100	94	187	79	...
tit_for_tat	100	101	103	106	...



Résultats classiques

- Résultats mis en avant notamment par AXELROD en 1984 à propos de `tit_for_tat`
- Possibilité d'apparition d'un ordre sans autorité extérieure
- Critères de qualité pour une stratégie (en évolution) :
 - bienveillance
 - réactivité
 - indulgence
 - simplicité (clarté du comportement)
- Les *bonnes* stratégies au dilemme classique le sont aussi dans les variantes

Nous remettons en cause la simplicité mise en avant par AXELROD

Principaux résultats de la thèse

Nous avons donc utilisé des simulations informatiques pour :

- vérifier la robustesse de notre adaptation
- étudier les dynamiques de populations
- évaluer de nouvelles stratégies
- essayer de créer automatiquement de *bonnes* stratégies

Nous avons montré formellement :

1. que certains panels de stratégies sont améliorables à l'infini
2. que dans le cas général les panels de stratégies ne sont pas améliorables à l'infini

Nous avons établi et étudié un modèle basé sur une légère modification du dilemme du prisonnier : le dilemme de l'ascenseur

⇒ De nouveaux arguments en faveur de la complexité dans le comportement

Robustesse

Mesure de robustesse du jeu :

- N stratégies classées dans un tournoi
- $P < N$ stratégies classées dans un sous-tournoi, chaque stratégie i est alors :
 - attendue à une position théorique C_i^t
 - classée à un position réelle C_i^r
- Combien de positions le changement de panel a fait évoluer :

$$\sum_{j=1}^{j=P} |C_{i,j}^t - C_{i,j}^r|$$

- Sur M sous-tournois choisis aléatoirement la moyenne est alors notre mesure de robustesse :

$$R_{(P/N),M,L} = \frac{\sum_{i=1}^{i=M} \left(\sum_{j=1}^{j=P} |C_{i,j}^t - C_{i,j}^r| \right)}{M}$$

$\Rightarrow R$ ne prend jamais de valeur supérieure à 3.

Dynamiques oscillatoires

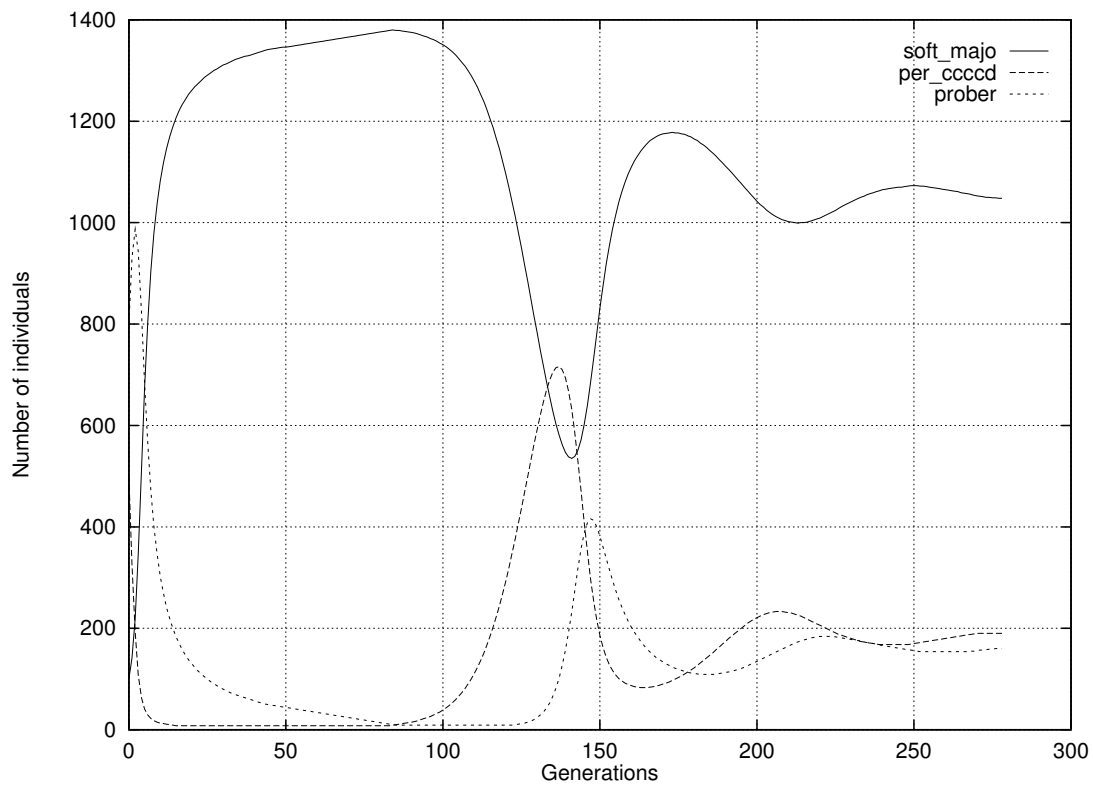
Nous avons recherchés systématiquement dans un grande nombre d'évolutions impliquant 3 stratégies des dynamiques non classiques.

Nous avons pu faire une classification des évolutions en 5 catégories :

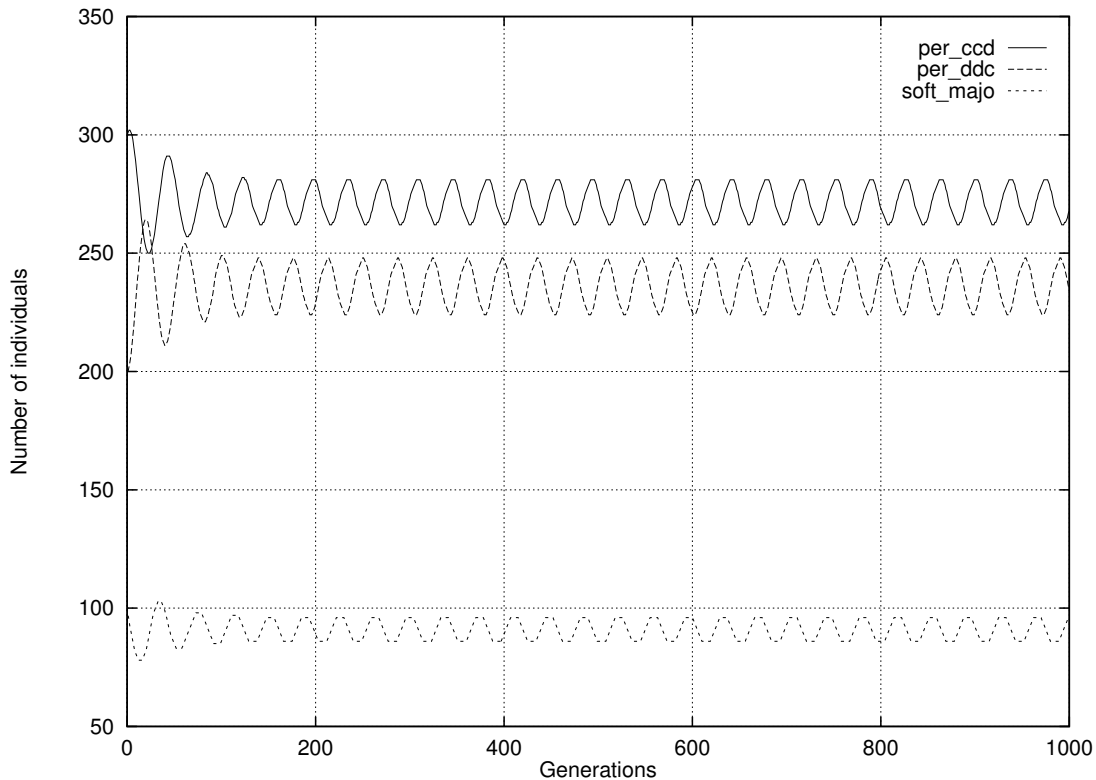
1. convergence monotone (99 % de nos expérimentations)
2. oscillations atténuées
3. mouvements périodiques
4. oscillations croissantes
5. oscillations désordonnées

⇒ Ces dynamiques *rare*s sont **très sensibles** aux modifications des conditions initiales d'expérimentation

Oscillations désordonnées

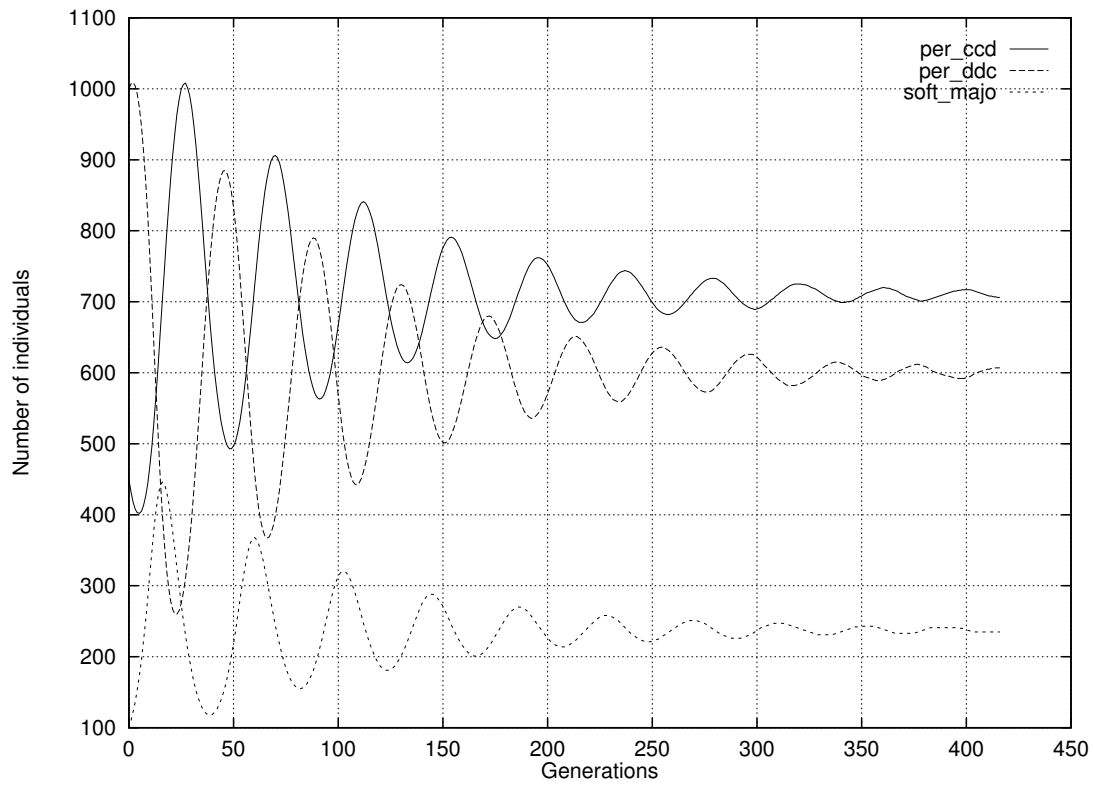


Mouvements périodiques

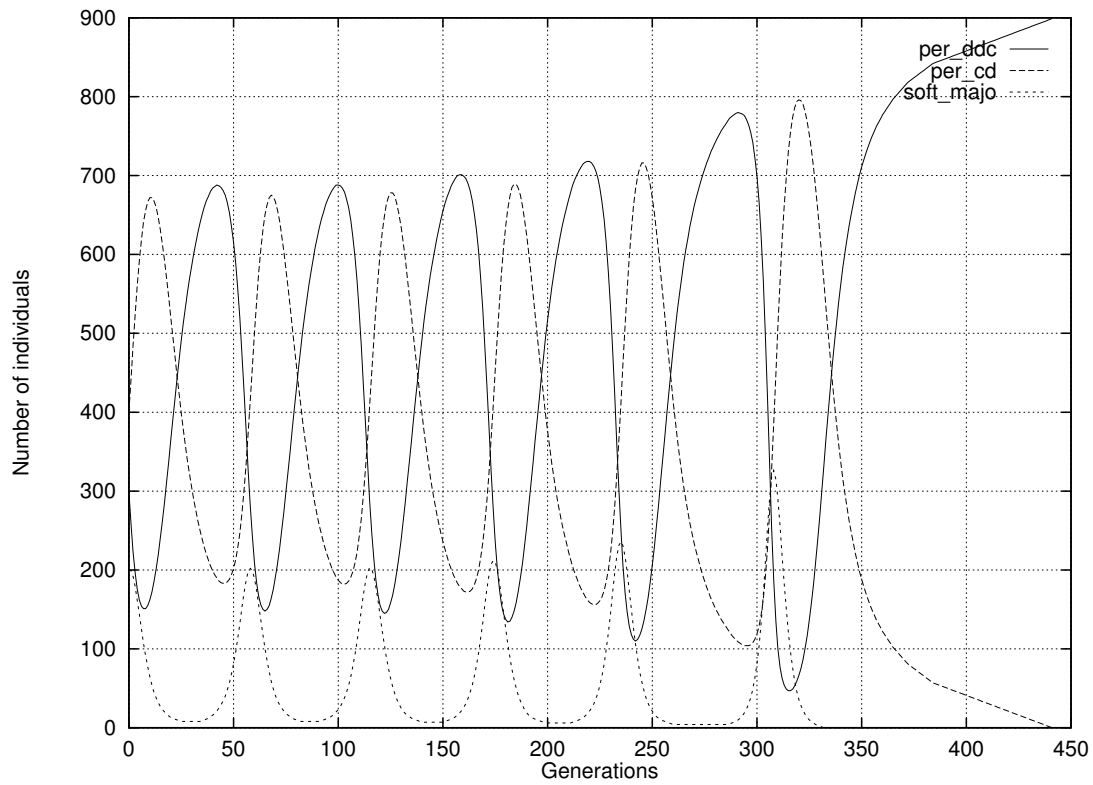


Les stratégies impliquées forment souvent un cycle en tournoi : A est meilleure que B, qui est meilleure que C qui est meilleure que A

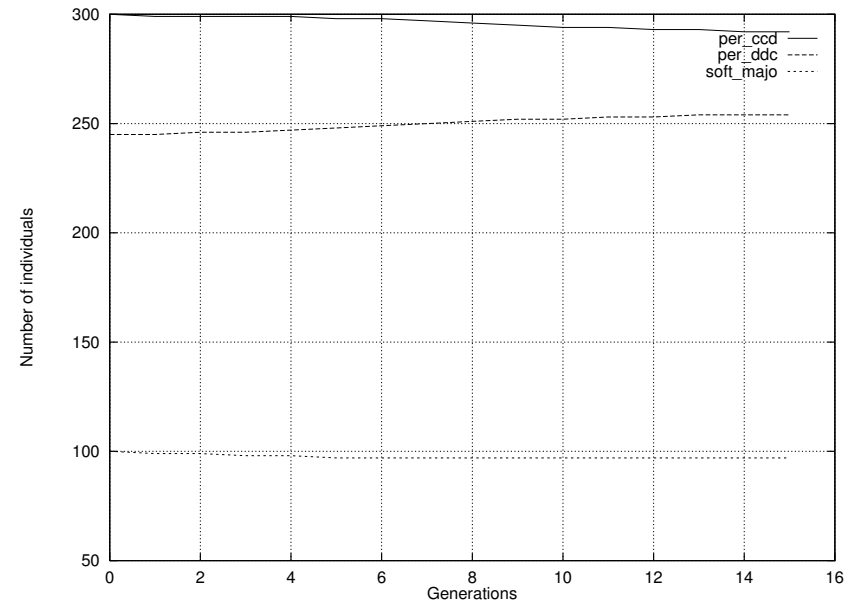
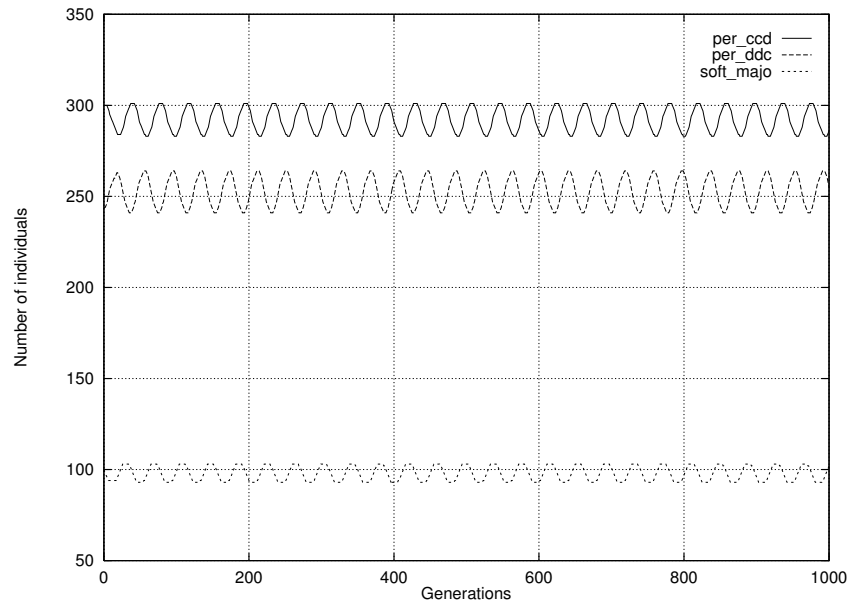
Oscillations atténuées



Oscillations croissantes



Sensibilité à la population initiale



Tous les paramètres sont identiques sauf la taille de la population initiale des per_ddc qui est de 244 à gauche et de 245 à droite

gradual

Première trahison de l'adversaire: $\boxed{D} \mapsto \boxed{D} \boxed{C} \boxed{C}$

Seconde trahison de l'adversaire: $\boxed{D} \mapsto \boxed{D} \boxed{D} \boxed{C} \boxed{C}$

Troisième trahison de l'adversaire: $\boxed{D} \mapsto \boxed{D} \boxed{D} \boxed{D} \boxed{C} \boxed{C}$

⋮

gradual coopère au premier coup, puis après la première trahison de l'adversaire trahit une fois et coopère deux fois, après la seconde trahison de l'adversaire trahit deux fois et coopère deux fois, ..., après la n^e trahison de l'adversaire trahit n fois et coopère deux fois.

bad_bet

- Coopérer jusqu'à ce que l'adversaire ne trahisse,
- Ensuite jouer successivement et pendant 4 coups chacun les stratégies :
 1. tit_for_tat
 2. all_c
 3. spiteful
 4. per_ccd
- Puis jouer pendant les 4 coups suivants la stratégie qui a rapporté le plus gros score pendant les 4 coups où elle a été utilisée.
- Recommencer ce choix tous les 4 coups.

Nouveaux critères

Les évaluations de `gradual` et `bad_bet` montrent qu'elles sont très supérieures à `tit_for_tat`. On obtient un *classement* :

$$\text{tit_for_tat} < \text{gradual} < \text{bad_bet}$$

Deux critères nouveaux semblent donc avoir été trouvés : au cours d'une partie une stratégie doit savoir :

- faire **évoluer** sa réaction aux trahisons de l'adversaire
- **adapter** cette réaction en fonction du comportement de l'adversaire

⇒ Ces deux critères accroissent la complexité des stratégies

Classes complètes (1)

- La définition objective d'un grand nombre de stratégies nécessite une méthode descriptive de construction d'un ensemble de stratégies
 - Pour définir un ensemble de stratégies on peut par exemple :
 - définir une structure capable d'être décodée en un comportement à adopter face à un adversaire
 - utiliser toutes les manières possibles de remplir cette structure comme autant de stratégies
- ⇒ approche génétique de la définition des individus

Classes complètes (2)

Considérons les stratégies qui ne se souviennent que de leur dernier coup et du dernier coup de leur adversaire.

On peut définir une de celles-ci par :

– le premier coup je joue

C

 puis

–

{	si au coup d'avant j'ai joué C et que lui a joué C alors je joue	<table border="1" style="display: inline-table; vertical-align: middle;"><tr><td>C</td></tr></table>	C
	C		
	si au coup d'avant j'ai joué C et que lui a joué D alors je joue	<table border="1" style="display: inline-table; vertical-align: middle;"><tr><td>D</td></tr></table>	D
	D		
si au coup d'avant j'ai joué D et que lui a joué C alors je joue	<table border="1" style="display: inline-table; vertical-align: middle;"><tr><td>C</td></tr></table>	C	
C			
si au coup d'avant j'ai joué D et que lui a joué D alors je joue	<table border="1" style="display: inline-table; vertical-align: middle;"><tr><td>D</td></tr></table>	D	
D			

On peut alors définir le *génotype* de cette stratégie comme :

C	C	D	C	D
---	---	---	---	---

⇒ 2^5 stratégies différentes

Classes complètes (3)

Plusieurs classes créées et utilisées :

- memory
- memory_with_dynamic_start
- memory_with_limited_dynamic_start
- binary
- binary_with_dynamic_start
- moore
- mealy

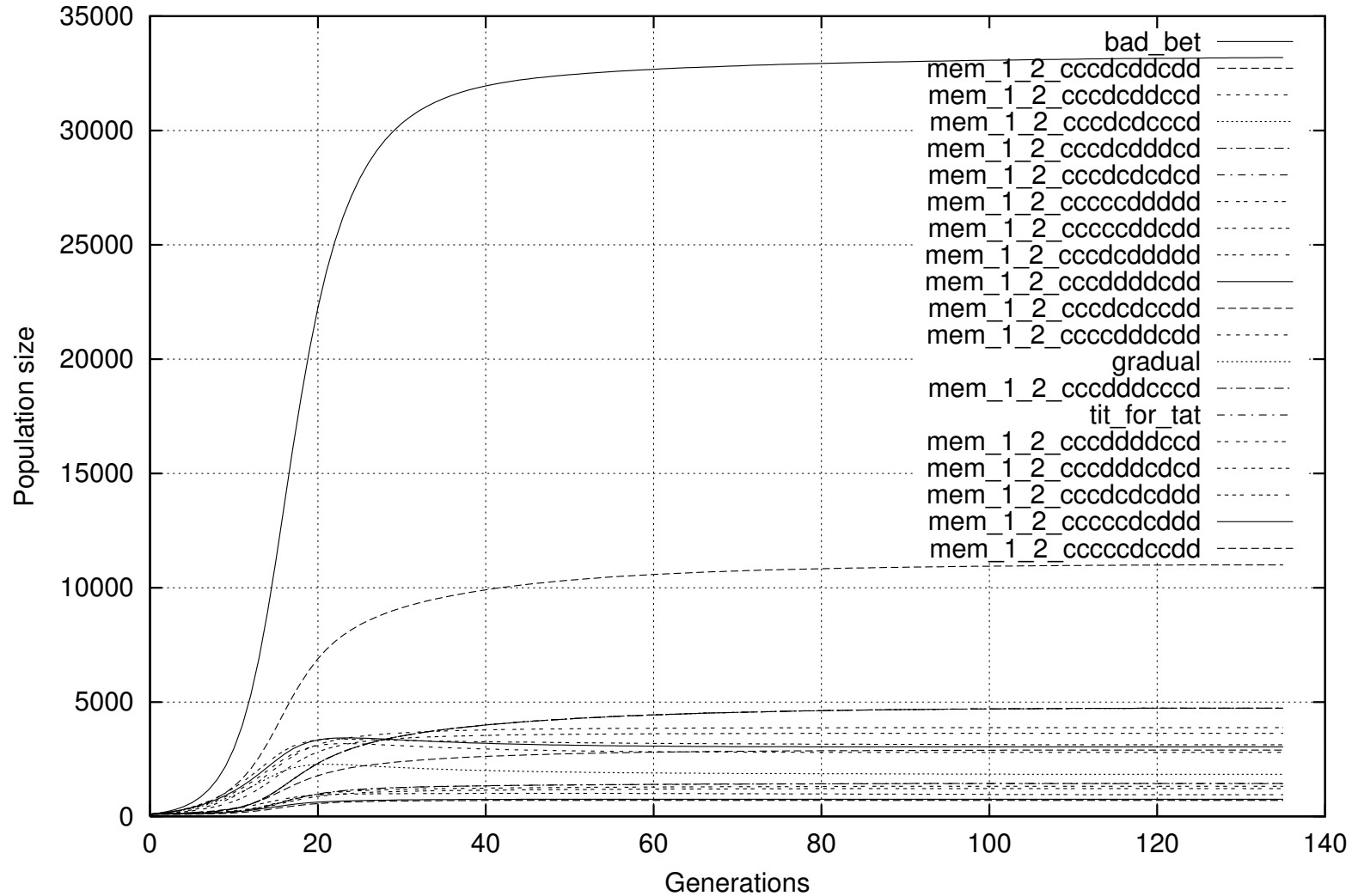
classe complète = évolution impliquant toutes les stratégies d'une classe

Les classes complètes permettent :

- d'évaluer des stratégies dans un environnement large
- de rechercher des bons comportements

Classes complètes (4)

mem_1_2+bad_bet+tit_for_tat+gradual



Classes complètes (5)

Classe	Taille	Meilleure	Génération	gradual	bad_bet
mem_0_1	8	mem_0_1_ccd	33	1	1
mem_0_2	64	mem_0_2_cccdcd	250	5	1
mem_1_1	32	mem_1_1_ccddd	20	2	1
mem_1_2	1 024	mem_1_2_cccdcdcd	175	10	1
mem_2_1	1 024	mem_2_1_dcccdcdddd	8 647		
memd_0_2	128	memd_0_2_cccccdcd	422		1
memd_1_2	2 048	memd_1_2_cccccdcdcd	75	15	1
memd_2_1	2 048	memd_2_1_dcccdcdddd	10 000		
memld_0_3	4 096	memld_0_3_ddcccccdcd	15 000		1
bin_0_1	32	bin_0_1_cddcc	57	1	1
bin_0_2	1 024	bin_0_2_dcddcdcccd	10 842		1
bin_1_1	512	bin_1_1_ccdddcdcc	70	1	11
bind_0_2	2 048	bind_0_2_cccccdcdcd	15 000		1
moore_0_1_2	128	moore_0_1_cd_00111	30	3	1
moore_1_1_2	2 048	moore_1_1_cd_001111101	31		1
mealy_0_1_2	1 024	mealy_0_1_c0c0c1c1d1	156	1	1

Conclusions

- Vérification des résultats du cas continu dans le cas discret
- De nouveaux arguments en défaveur de la simplicité des comportements :
 - dynamiques non monotones
 - critère de qualité à complexité croissante :
 - évolution des réactions
 - adaptations des réactions
 - mauvais comportement de `tit_for_tat` dans des évolutions de grande ampleur
- Tous les travaux sont reproductibles aisément (simulateur logiciel disponible librement)

Perspectives

- Essayer de faire des simulations encore plus grandes (classes complètes)
- Poursuivre l'étude du dilemme itéré de l'ascenseur
- Diffusion des résultats, via entre autre le web, pour essayer de trouver des vérifications plus *pratiques* de nos résultats
- Étude de modèles moins simples, et plus abstraits de coopération