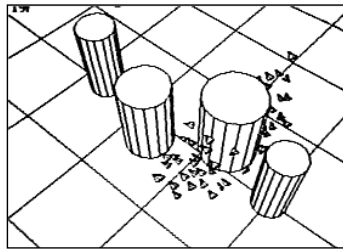


# Comportements d'agents virtuels



## Vie artificielle, robotique et jeux vidéos

Nicolas Bredèche

Professeur des Universités (ISIR, SU)

[nicolas.bredeche@upmc.fr](mailto:nicolas.bredeche@upmc.fr)

Module: 2i013

Dernière mise à jour: 2019-02-04

--

- Modélisation de comportements
  - Objectif : comportement autonome dans un monde ouvert
  - Liens avec:
    - Vie Artificielle
    - Architecture de contrôle en robotique
    - Jeux vidéos

# Fourmis de Langton

- Modélisation individu-centrée
  - Fourmi de Langton et Turmites : structure émergente
- Modèle computationnel
  - Modélisation et simulation de la propagation d'un virus

- Principes
  - La simulation discrète permet de simuler des agents *localisés*
  - La mise à jour est faite de manière *distribuée* pour chaque agent
  - On peut capturer les particularités environnementales et comportementales
- Applications:
  - Toute unité capable de se déplacer (molécules, individus, groupes, ...)

## Automate cellulaire $\neq$ simulation agent-centrée

- Automate cellulaire, c'est...
  - un automate 2D
  - une tête de lecture/écriture
  - état(s) interne(s)
- Une simulation basée agent, c'est...
  - un ou plusieurs agent(s)
    - ▶ chaque agent est localisé
    - ▶ chaque agent peut avoir des états internes
  - un environnement
    - ▶ dimensions : 0D ("soupe"), 1D, 2D, 3D... ou plus
    - ▶ dynamique ou non

# Fourmi de Langton

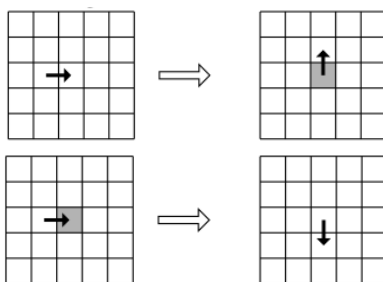


[Langton, 1986]

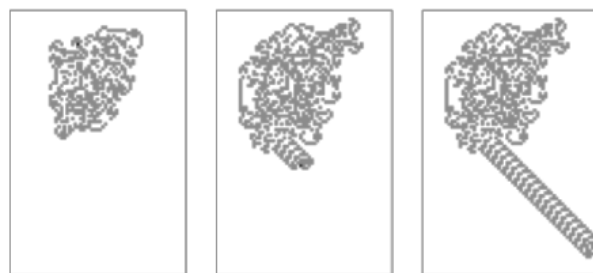
- Un «agent»
  - Un état interne : orientation
  - Des actions possibles : rotation, avancer, changer l'état d'une case
- Un environnement (ie. ensemble de cases)
  - Deux états: noir ou blanc (présence de phéromone ou non)
- Deux actions:
  - Si [blanc] alors: inverse état, tourne à 90° vers la gauche, avance
  - Sinon ([noir]) : inverse état, tourne à 90° vers la droite, avance

source: wikipedia

# Fourmi de Langton



Règles au niveau microscopique



Emergence au niveau macroscopique

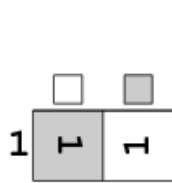
## Emergence

Apparition d'une nouvelle caractéristique (structurelle ou comportementale) observable à une échelle macroscopique et résultant de phénomènes agissant au niveau microscopique.

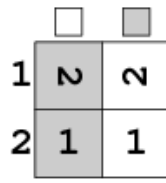
Ici: convergence vers une stratégie comportementale particulière

# “Turmites”

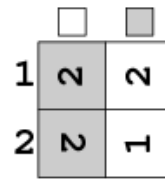
“Turing termites”



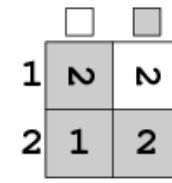
turmite à 1 état  
(miroir de la fourmi  
de Langton)



turmites à 2 états  
exemple 1



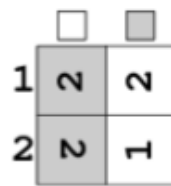
turmites à 2 états  
exemple 2



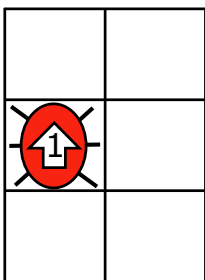
turmites à 2 états  
exemple 3

source: <http://mathworld.wolfram.com/Turmite.html>

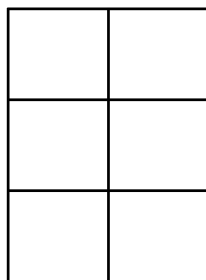
## Turmite à deux états en environnement “toroïdal”



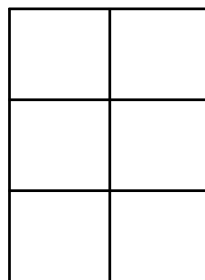
environnement  
**toroïdal**



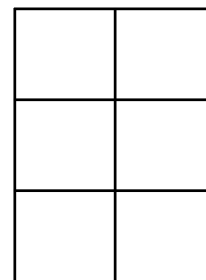
t=0



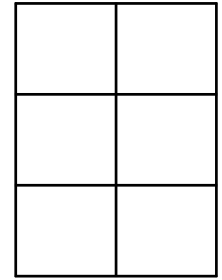
t=1



t=2





t=3



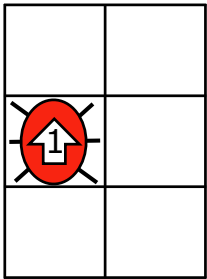
t=4

Exercice: sur un environnement 3x2 partant du centre,  
dérouler les 4 premières itérations de cette turmite.

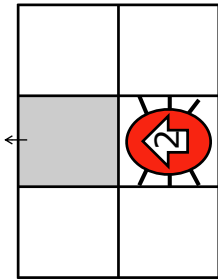
# Turmite à deux états en environnement "toroïdal"

		
1	2	2
2	2	1

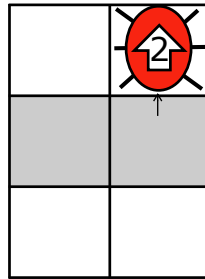
environnement  
**toroïdal**



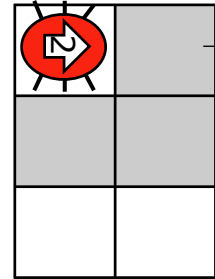
t=0



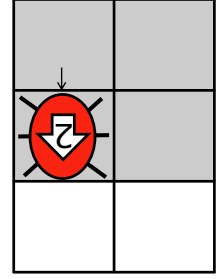
t=1



t=2



t=3



t=4

Epidémiologie  
le modèle SIR

# Epidémiologie

- Le modèle SIR
  - S: sain
  - I: infecté
  - R: guéri (ou mort)
- Modélisation mathématique [Kermack, McKendrick, 1927]

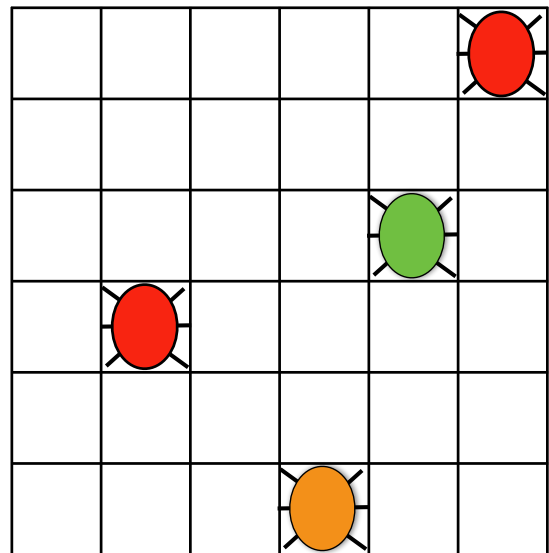
$$\begin{aligned}\frac{dS}{dt} &= -\frac{\beta SI}{N} \\ \frac{dI}{dt} &= \frac{\beta SI}{N} - \gamma I \\ \frac{dR}{dt} &= \gamma I\end{aligned}$$

Résolution analytique triviale si on considère une soupe, mais que se passe t'il si les agents sont localisés?

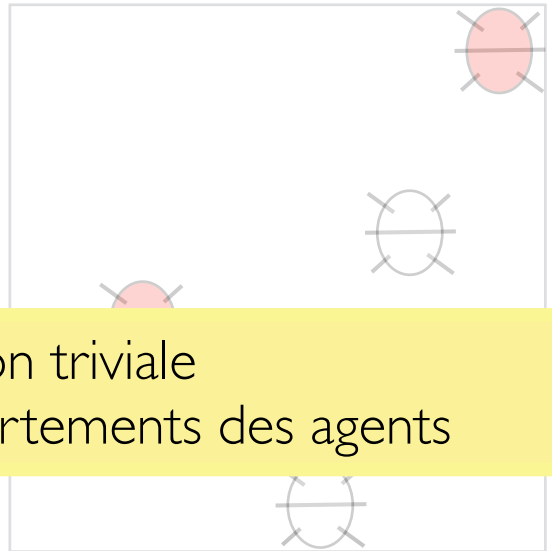
source équations: wikipedia

## Propagation de Virus (1/2)

- Population d'agents
  - état: sain, infecté ou guéri
  - actions: N,S,E,W
  - un agent par case
- Règles:
  - Infection: un agent infecté contamine les agents voisins
  - Guérison: après N itérations



# Propagation de Virus (1/2)



Modélisation non triviale  
Il faut modéliser les comportements des agents

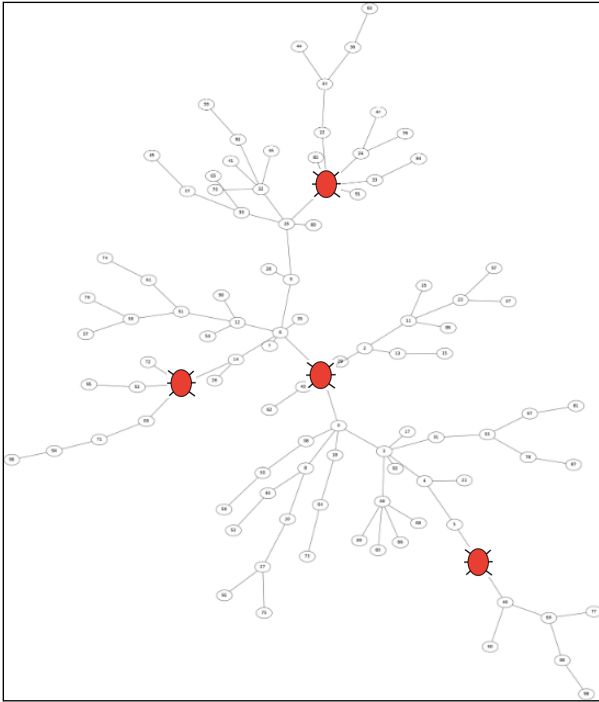
- actions: N,S,E,W
- un agent par case
- Règles:
  - Infection: un agent infecté contamine les agents voisins
  - Guérison: après N itérations sans ré-infection

DEMO  
epidemiology.py

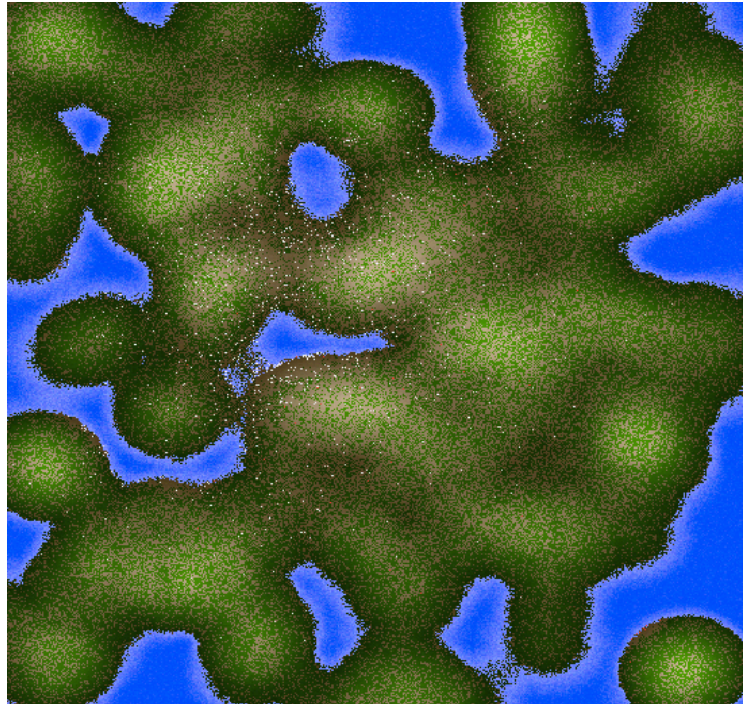
Effet de la densité  
Effet de la stochasticité



# Propagation de Virus (2/2)

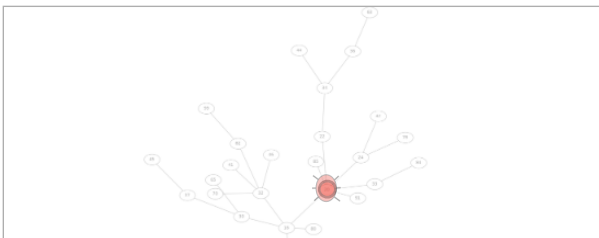


Impact de la topologie

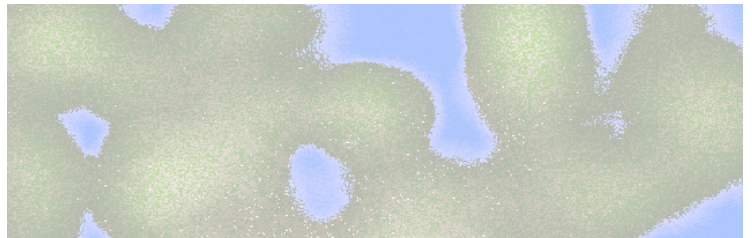


Impact de l'environnement

# Propagation de Virus (2/2)



Impact de la topologie



Impact de l'environnement

Modélisation non triviale  
Il faut modéliser la topologie de l'environnement

# Architecture de comportements

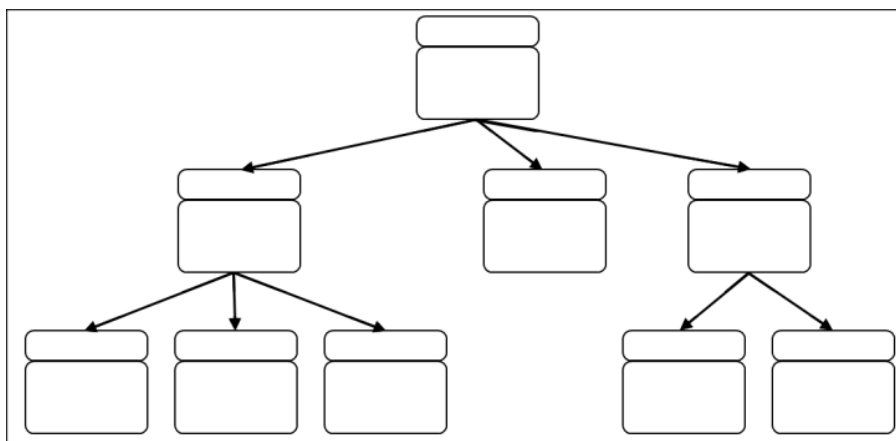
- Objectifs
  - Combiner des comportements plus simples
  - Sélectionner le comportement adéquat selon le contexte
- Modélisation
  - Règles, arbres, automates, etc.

## Behavior trees

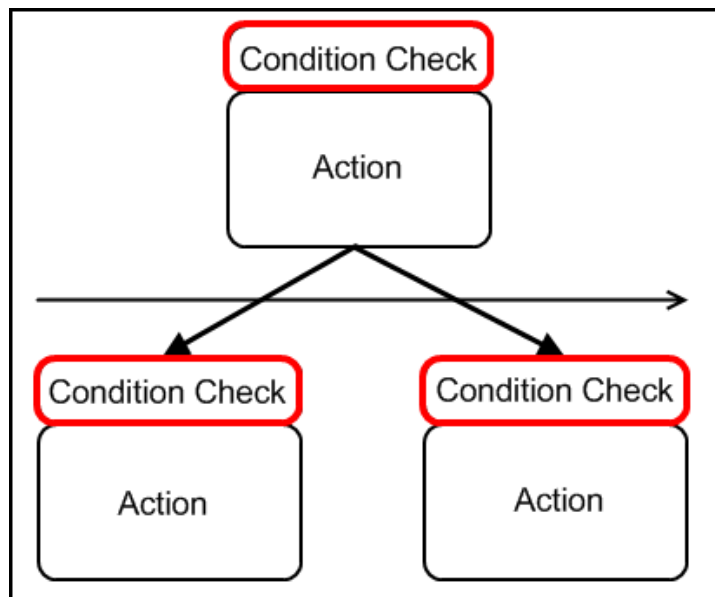
nicolas.bredeche@upmc.fr

# “Behavior Tree”

Arbre de comportements



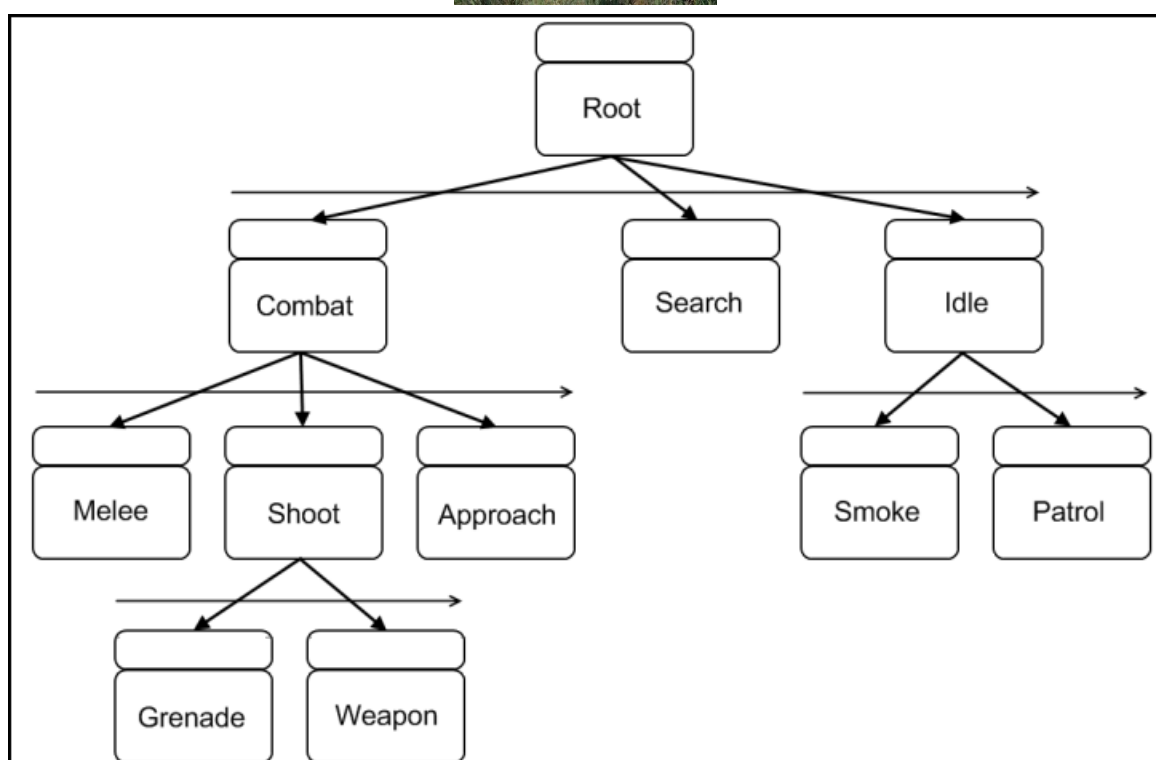
- Facile à comprendre, facile à modifier
- Modification temps réel possible
- Duplication de sous arbres facile



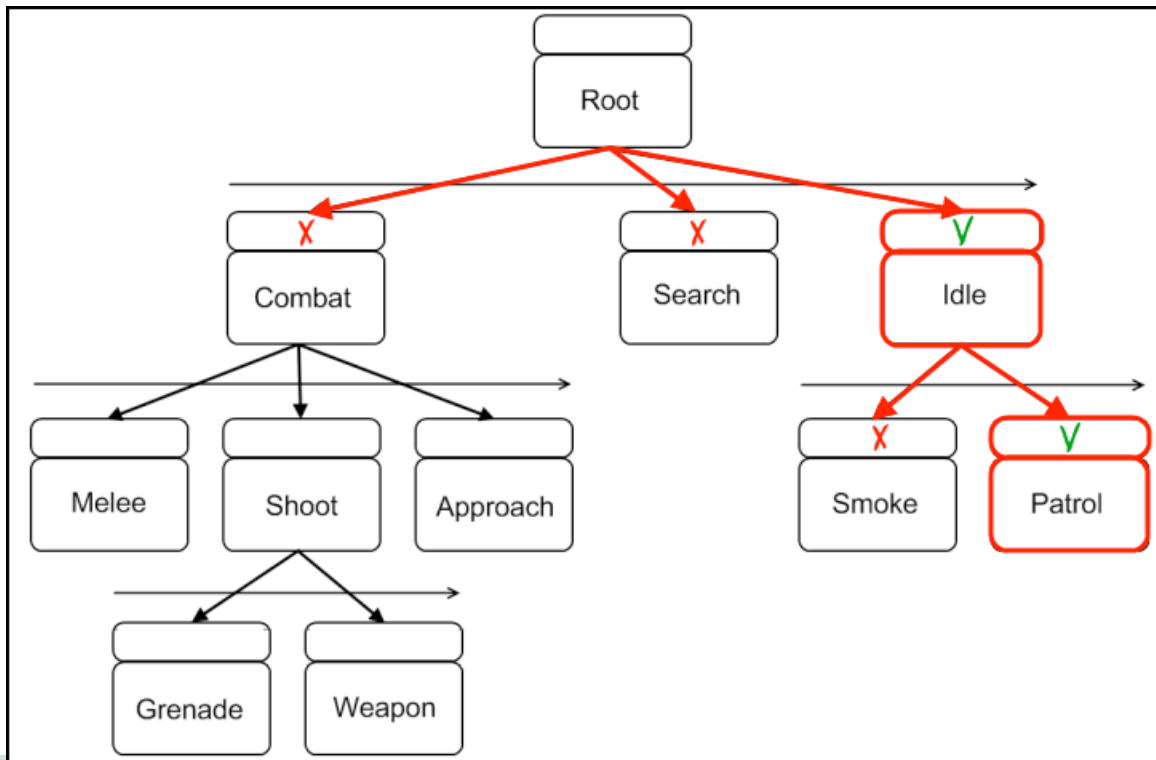
- Un noeud = condition + action (la condition est validée)
- Ordre et séquence (les actions aux feuilles peuvent être enchainées)
- Comportement défini au niveau des feuilles de l'arbre

source: Pillosu, Algamedev Paris 2009

Crysis (exemple)

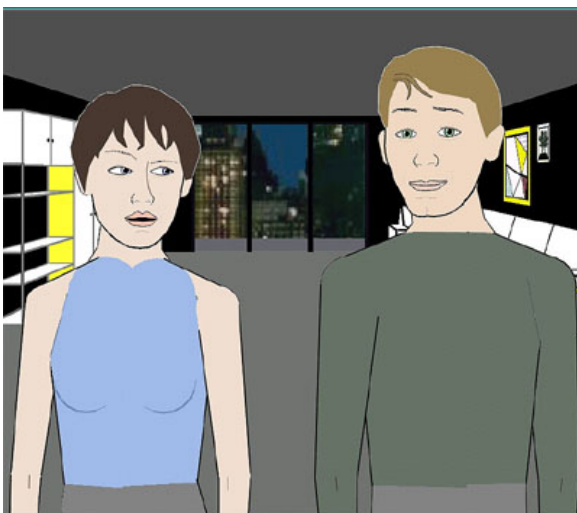


source: Pillosu, Algamedev Paris 2009

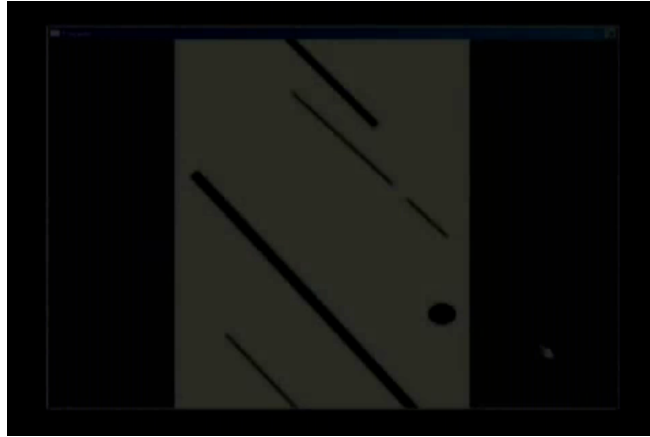


source: Pillosu, Algamedev Paris 2009

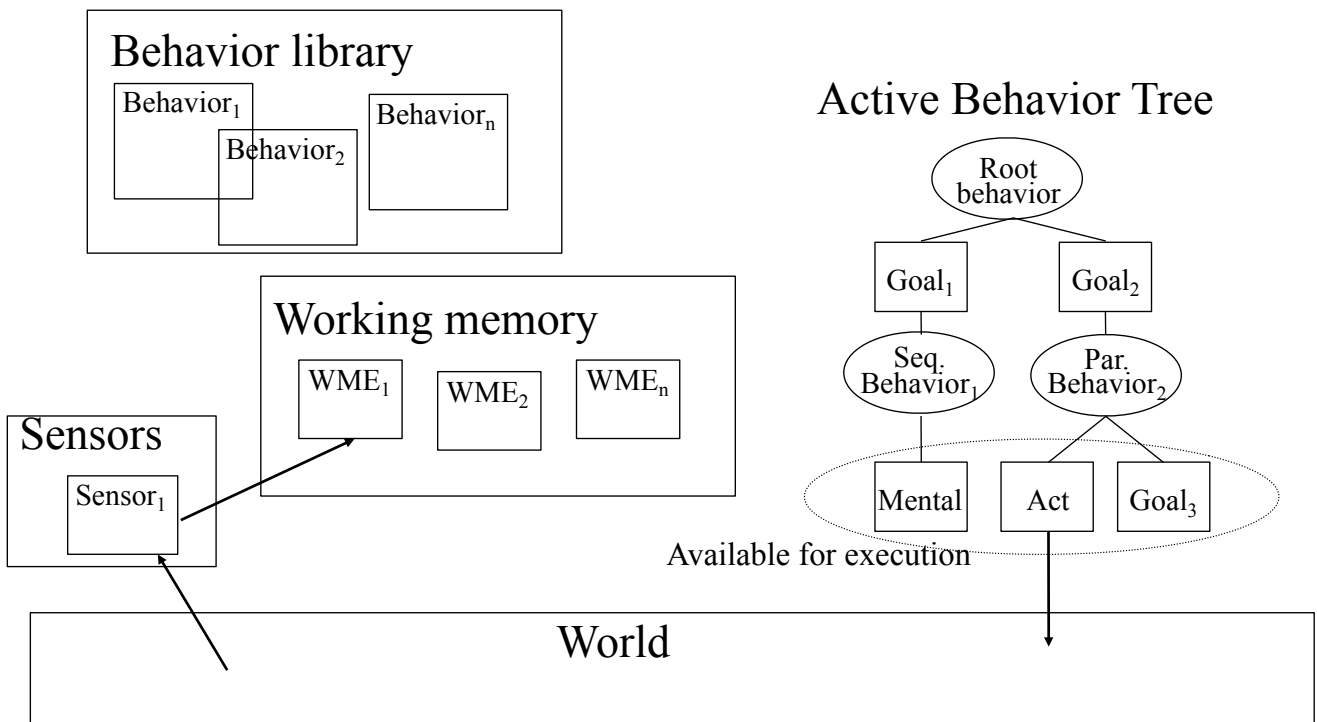
## Façade



- Dramatic world inhabited by computer controlled characters (believable agents)
- The user (player) plays a protagonist within the story, first-person point of view
- The player experiences a story with a dramatic arc



# Façade



## Etude de cas: PACMAN

Les images des transparents suivants sont issues du site « **The Pacman Dossier** » de Jamey Pittman

Source: <http://home.comcast.net/~jpittman2/pacman/pacmandossier.html> (au 2016-2-2, cette page n'est pas accessible)

Une copie de ce dossier se trouve ici: [http://www.gamasutra.com/view/feature/3938/the\\_pacman\\_dossier.php](http://www.gamasutra.com/view/feature/3938/the_pacman_dossier.php)

nicolas.bredeche@upmc.fr

### PACMAN



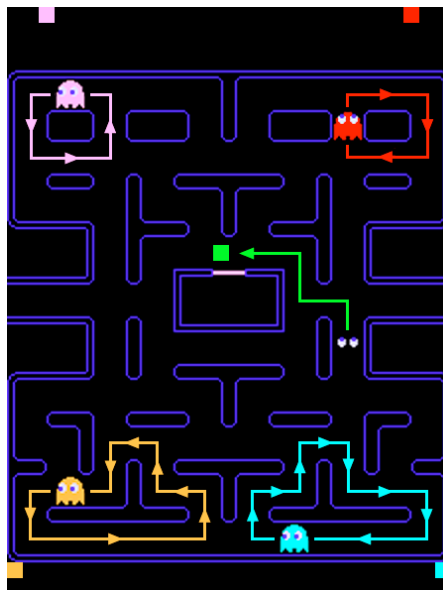
## PACMAN



Mode	Level 1	Levels 2-4	Levels 5+
Scatter	7	7	5
Chase	20	20	20
Scatter	7	7	5
Chase	20	20	20
Scatter	5	5	5
Chase	20	1033	1037
Scatter	5	1/60	1/60
Chase	indefinite	indefinite	indefinite

Source: <http://home.comcast.net/~jpittman2/pacman/pacmandossier.html>

## PACMAN



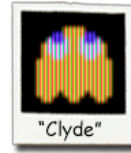
Comportements de dispersion  
 (« scatter », chaque fantôme à une zone de confort définie)

en vert: comportement « mort »

Source: <http://home.comcast.net/~jpittman2/pacman/pacmandossier.html>

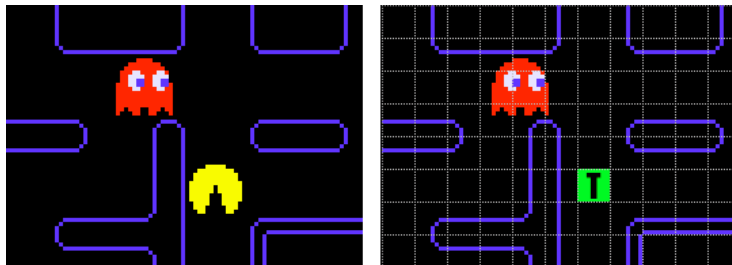
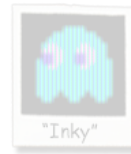
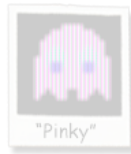


## PACMAN



Source: <http://home.comcast.net/~jpittman2/pacman/pacmandossier.html>

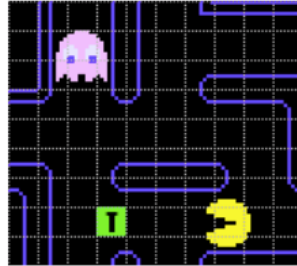
## PACMAN



Cible: pacman

Source: <http://home.comcast.net/~jpittman2/pacman/pacmandossier.html>

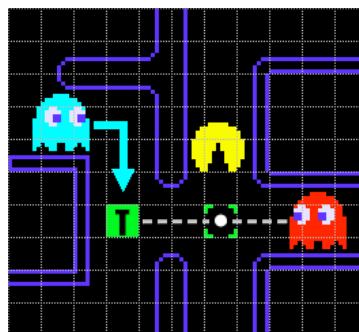
## PACMAN



Cible: 4 cases devant pacman

Source: <http://home.comcast.net/~jpittman2/pacman/pacmandossier.html>

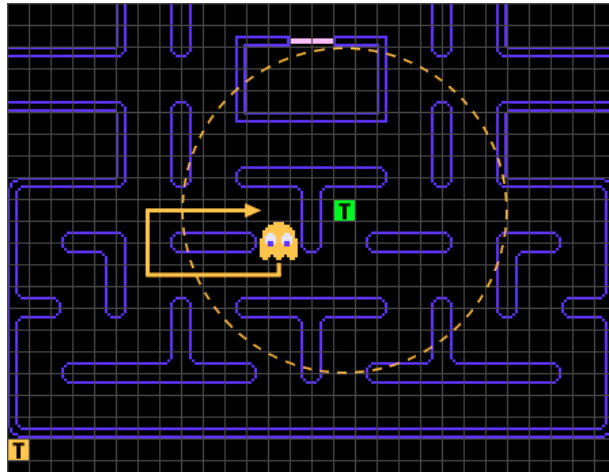
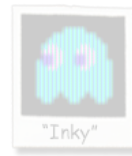
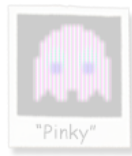
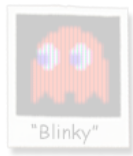
## PACMAN



Cible:  
repère: 2 cases devant pacman  
double le vecteur (rouge,repère)

Source: <http://home.comcast.net/~jpittman2/pacman/pacmandossier.html>

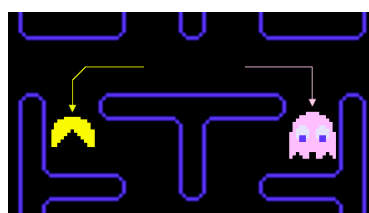
# PACMAN



Cible:  
si pacman est à plus de 8 cases: chasse comme « blinky »  
sinon: rejoint sa zone de confort (en bas, à gauche)

Source: <http://home.comcast.net/~jpittman2/pacman/pacmandossier.html>

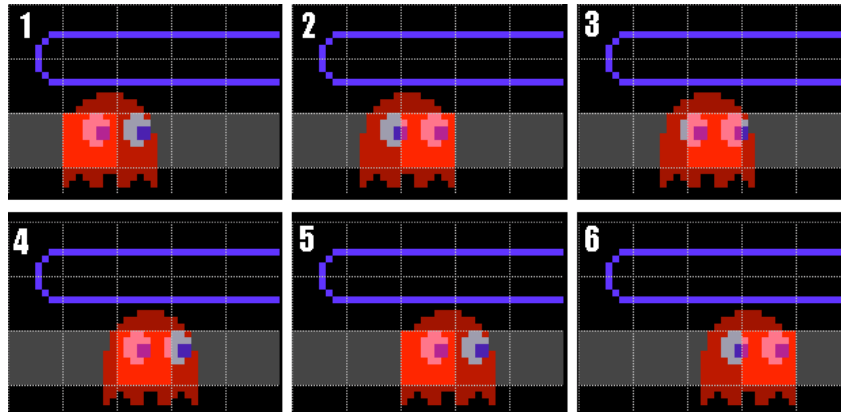
# PACMAN



Pacman tourne plus vite!

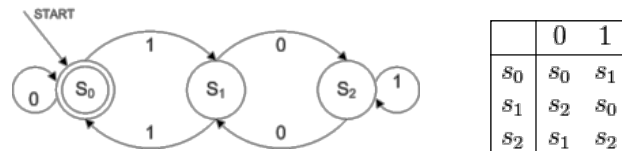
Source: <http://home.comcast.net/~jpittman2/pacman/pacmandossier.html>

Rendu vs. état du monde: dans quelle case suis-je ?



4 animations pour une même case

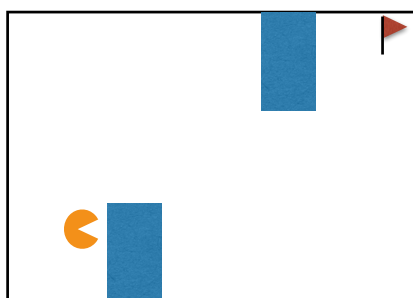
Source: <http://home.comcast.net/~jpittman2/pacman/pacmandossier.html>



# Conclusions

- Des comportements **simples** peuvent générer des dynamiques individuelles et collectives **complexes**
- Plus les comportements sont **complexes** (arbres, mémoire, etc.), plus les interactions (entre agents, avec l'environnement) sont **difficiles à prévoir**

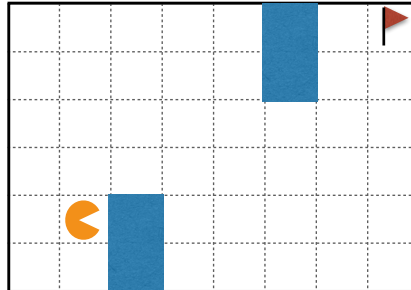
# Recherche de chemins



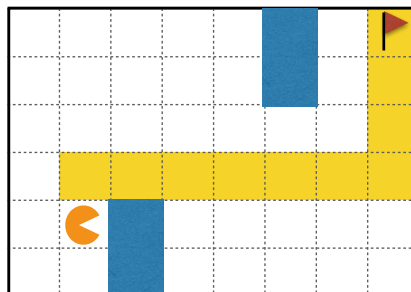
Comment atteindre le drapeau?

## Recherche avec une méthode naïve

hypothèse: il suffit de se rapprocher de la cible



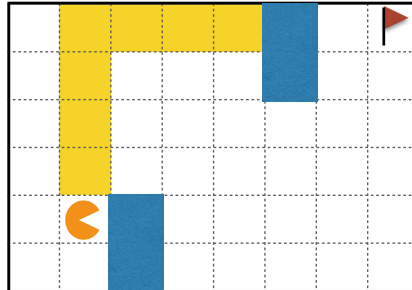
## Recherche avec une méthode naïve



cas 1: déplacement NSEW  
avec préférence déplacement horizontal

succès!

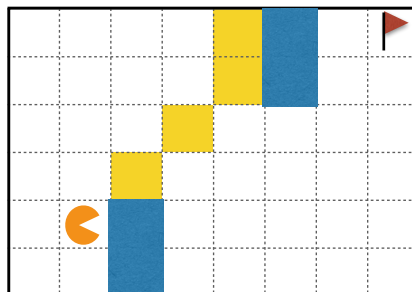
## Recherche avec une méthode naïve



cas 2: déplacement NSEW  
avec préférence déplacement vertical

bloqué!

## Recherche avec une méthode naïve



cas 3: déplacement 8 directions

bloqué!





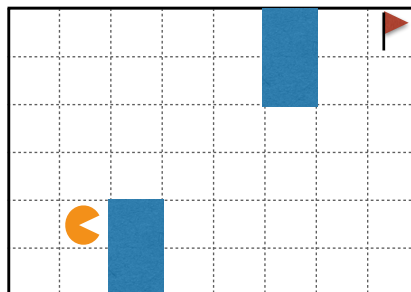
Dans ce labyrinthe, pas de blocage possible pour Blinky

Je ne suis pas très intelligent, mais ça suffit!

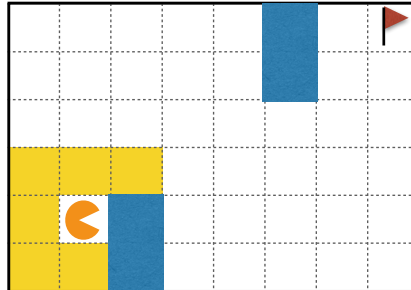


## Recherche en largeur d'abord

hypothèse: direction dans les 8 directions possibles

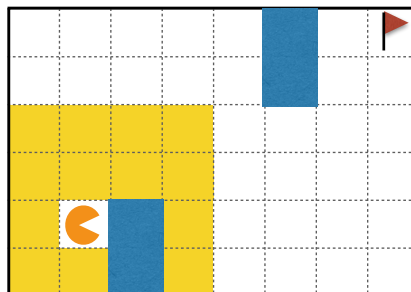


## Recherche en largeur d'abord



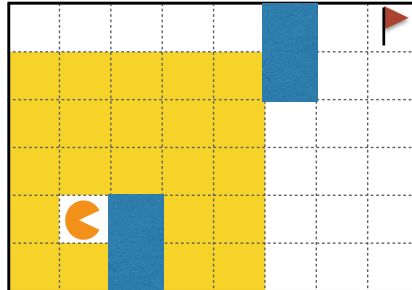
explore toutes les cases autour de la case de départ

## Recherche en largeur d'abord



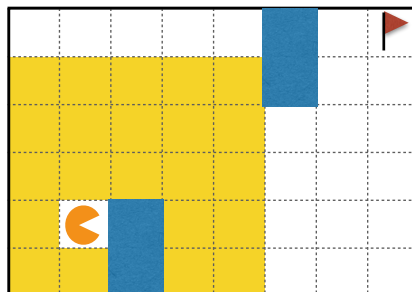
pour toutes les cases explorées, explore les cases voisines

## Recherche en largeur d'abord

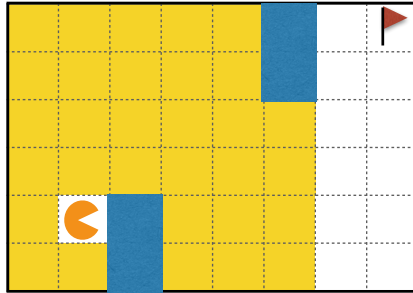


etc.

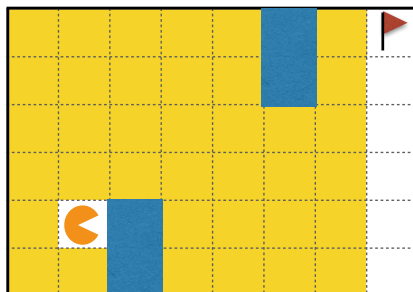
## Recherche en largeur d'abord



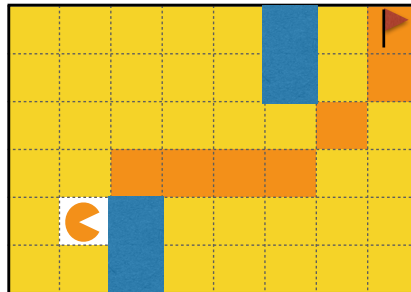
## Recherche en largeur d'abord



## Recherche en largeur d'abord



## Recherche en largeur d'abord

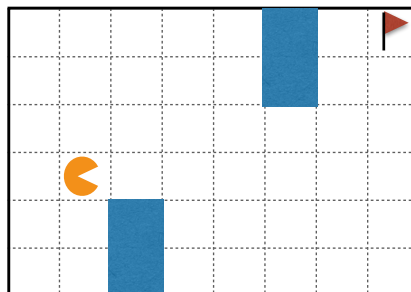


Exemple dans le pire cas  
(mais cela dépend du problème)

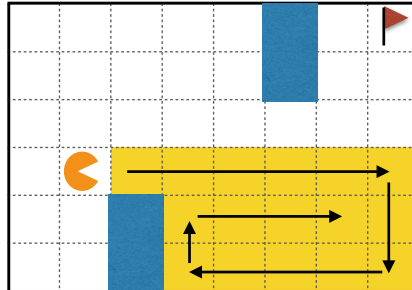
succès!

## Recherche en profondeur d'abord

hypothèse: direction dans les 8 directions possibles

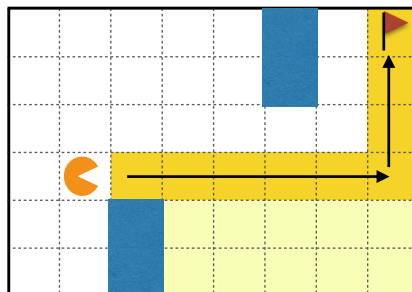


## Recherche en profondeur d'abord



explore dans une direction choisie au hasard jusqu'à ne plus pouvoir se déplacer sans repasser sur une case déjà visitée

## Recherche en profondeur d'abord



reprend l'exploration à partir de la dernière case avec une voisine pas encore explorée

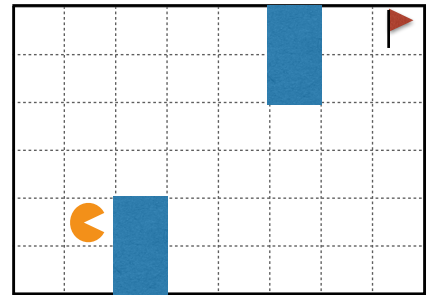
succès!

- Méthodes exhaustives

- Largeur d'abord

- Profondeur d'abord

➡ potentiellement très coûteuse



- Méthodes "naïves"

- Avance vers la cible. Si obstacle, suit le contour de l'objet

- Avance vers la cible. Si obstacle, recule (direction et nombre de cases aléatoires)

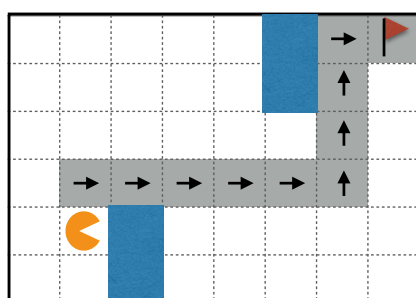
➡ blocage possible, mais peut exploiter des régularités (cf. pacman)

- Méthodes heuristiques

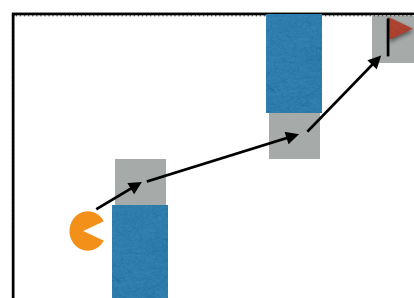
- Recherche best-first (nécessite une heuristique pour évaluer les déplacements)

- Algorithme A\* (extension de Dijkstra, c'est une sorte de best-first)

- Améliorations de A\* (D\*, IDA\*, Fringe, FSA\*, GAA\*, LPA\*, SMA\*, Jump point search, Theta\*, ..., UCT)



exploration case par case



définition de waypoints

#### Quelques pointeurs

<http://en.wikipedia.org/wiki/Pathfinding>

<http://gamedev.stackexchange.com/questions/28041/path-finding-algorithms>

[http://fr.wikipedia.org/wiki/Algorithme\\_A\\*](http://fr.wikipedia.org/wiki/Algorithme_A*)

# Synthèse

## Synthèse

- Ce que l'on a vu
  - Comportements réactifs simples
  - Comportements complexes, avec et sans mémoire
  - Recherche de chemins
- Ce qu'il faut retenir
  - Comment concevoir des comportements
    - ▶ Fourmi de Langton, propagation de Virus
    - ▶ Base de règles, arbres, graphes de comportements
    - ▶ Heuristiques comportementales