Jeu et moteur physique

Présentation ISN 2015 - 12/03/2015

Vincent THOMAS – université de lorraine
Vincent.thomas@loria.fr

http://webloria.loria.fr/~vthomas/page_ext/ISN_2015/

Point de départ

- Jeux et animation interessent étudiants
- Difficile envisager le continu (ex Mario)
  - animation (temps continu)
  - deplacement (espace continu)
- Mais une fois les ficelles montrées, c'est trés facile d'accés

Objectif de cet atelier

- Faire un moteur physique **basique** (mécanique du point) est trés simple.

- Un moteur physique **basique** permet de faire beaucoup de choses.

- Un moteur physique **basique** peut être une base pour de nombreux projets accessibles pour des èleves de terminale.

Contenu

- En amateur
  - Physique longtemps
  - Projets tutorés

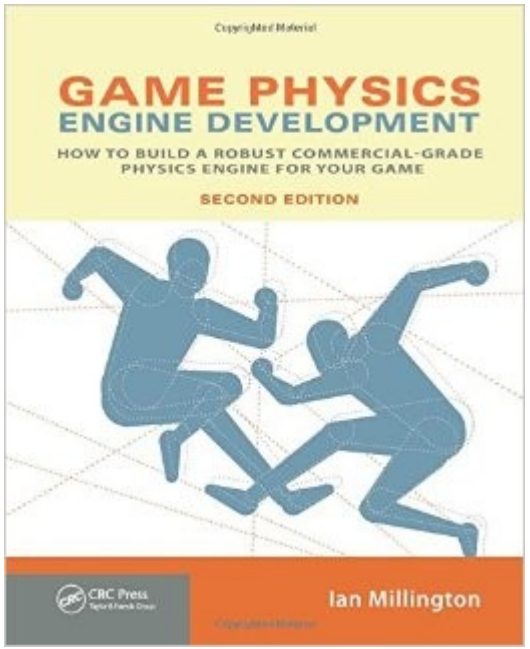
- Reference
  - Tutoriel Siggraph 2001 "pixar animation"

http://www.pixar.com/companyinfo/research/pbm2001/index.html

Contenu

- En amateur
  - Physique longtemps
  - Projets tutorés

- Reference
  - "Game physics engine dev"
    - Millington (2007 – reed 2010)

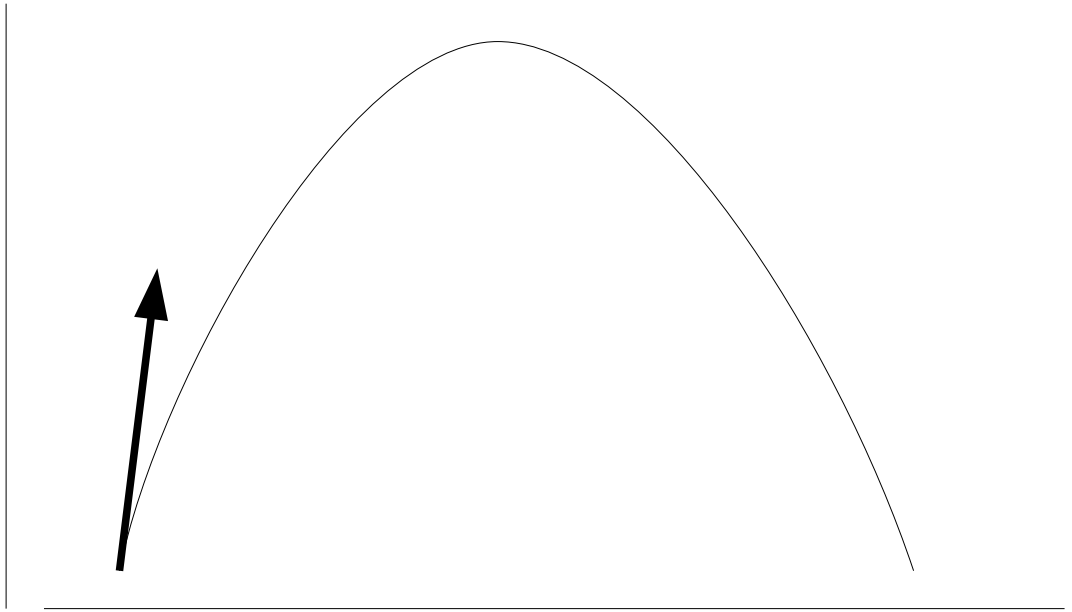


Plan

- Mecanique du point
- Moteur de jeu
- Mecanique du point (bis)
- Comportements collectifs
- Steering behaviour
- Probleme de controle

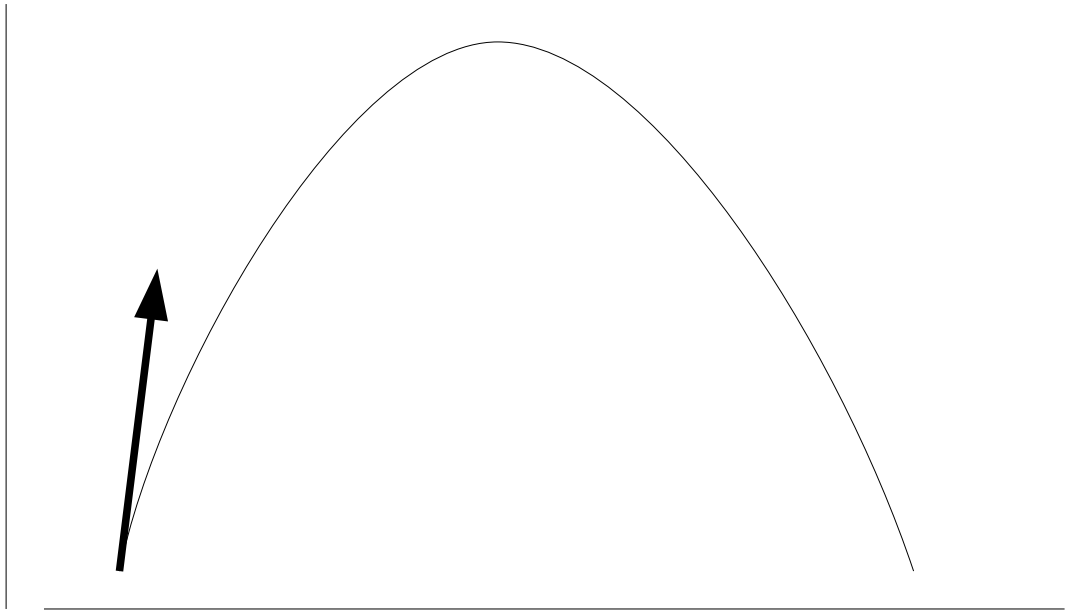
Physique du solide

- Gravité constante



Physique du solide

- Gravité constante
  - Erreurs d'approximation

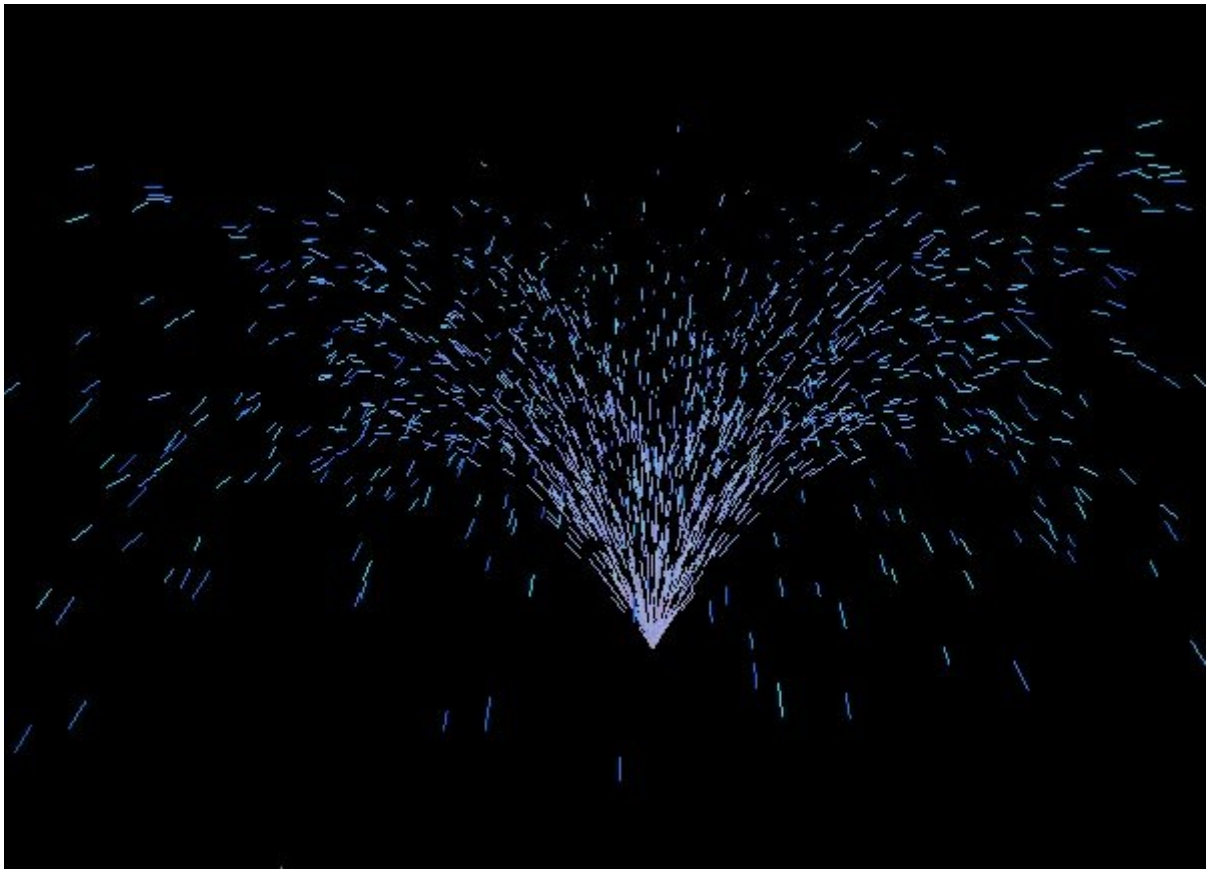


Montrer exemples

moteur physique sans graphique

Physique du solide

- Gravité constante
  - Systeme de particules



http://www.darwin3d.com/gamedev/articles/col0798.pdf

Plan

- Mecanique du point
- Moteur de jeu
- Mecanique du point (bis)
- Comportements collectifs
- Steering behaviour
- Probleme de controle

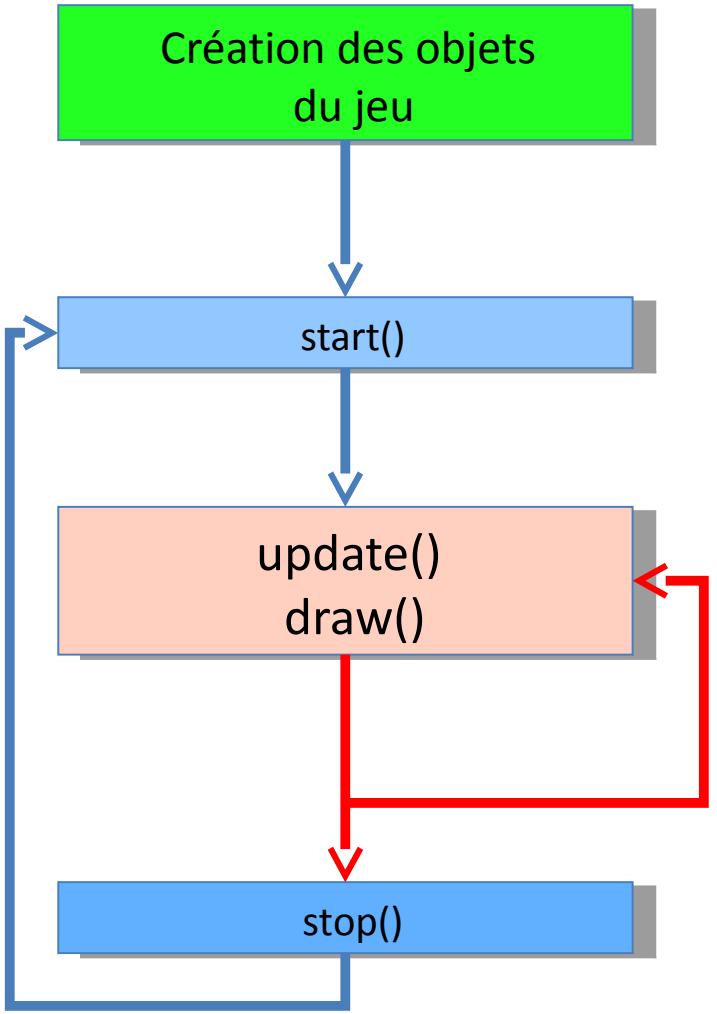
Moteur de jeu

- Éléments
  - Boucle de jeu
  - Gestion du temps
  - Gestion de l'affichage
  - Gestion du controle

- Possibilité
  - Construire from scratch
  - Utiliser libraries
    - Python : pygame, ...
    - Java: slick2D, ...

Moteur de jeu

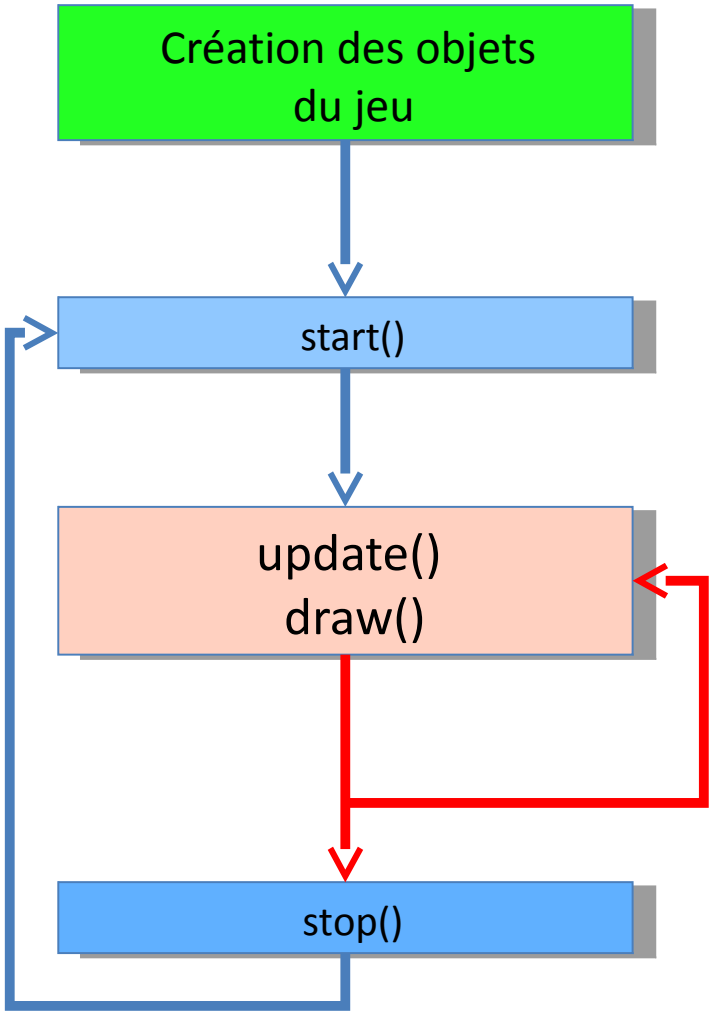
- Boucle de jeu



Boucle principale
du jeu

Moteur de jeu

- Boucle de jeu

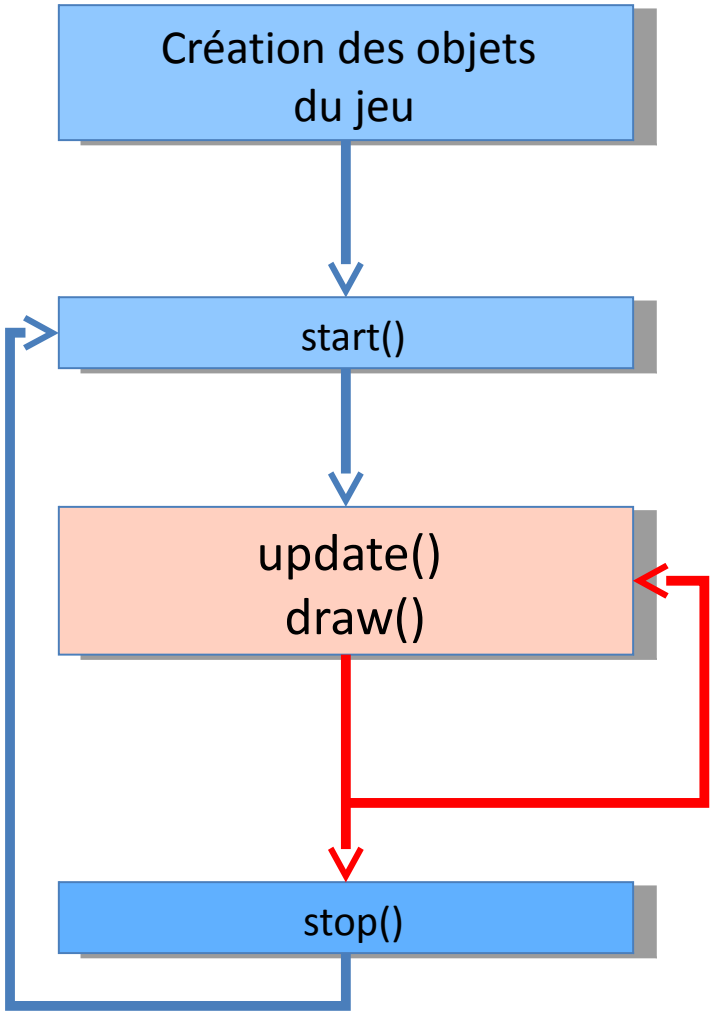


Cf fichier test

Boucle principale
du jeu

Moteur de jeu

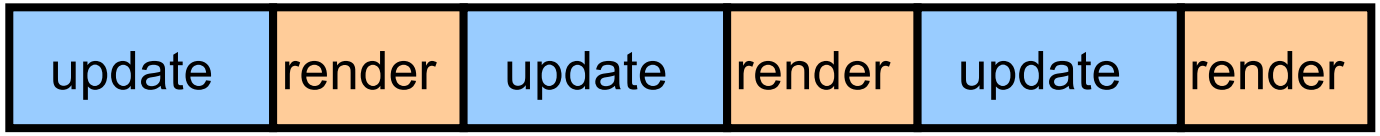
- pygame



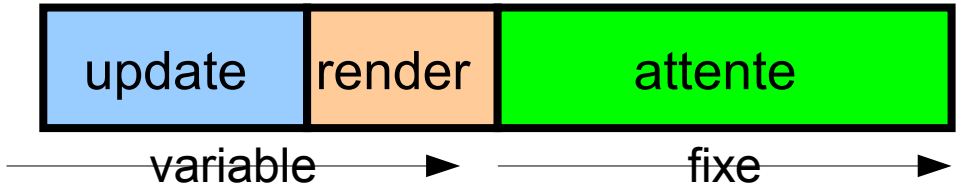

```
#on cree un jeu (approche objet)
jeu=Jeu()
# jeu_fini est un boolean qui précise quand le jeu est fini
jeu_fini = False

# on creer une horloge pour réguler la vitesse de la boucle de jeu
clock = pygame.time.Clock()

# -------- Boucle principale -----------
# tant que le jeu n'est pas fini
while not jeu_fini:

    # --- on traite les evenements
    jeu.traiter_evenement()

    # --- on fait evoluer le jeu
    jeu.evoluer()

    # --- on dessine le jeu
    jeu.dessiner()
    pygame.display.flip()

    # --- on demande d'attendre ce qu'il faut pour un FPS de 60
    clock.tick(60)
# -------- Fin Boucle principale -----------

pygame.quit()
```

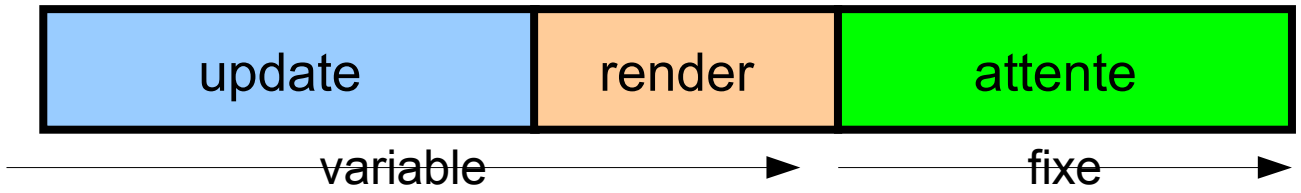
Moteur de jeu

- Gestion du temps
  - Sans attente
  - Probleme: vitesse du jeu indexée sur machine



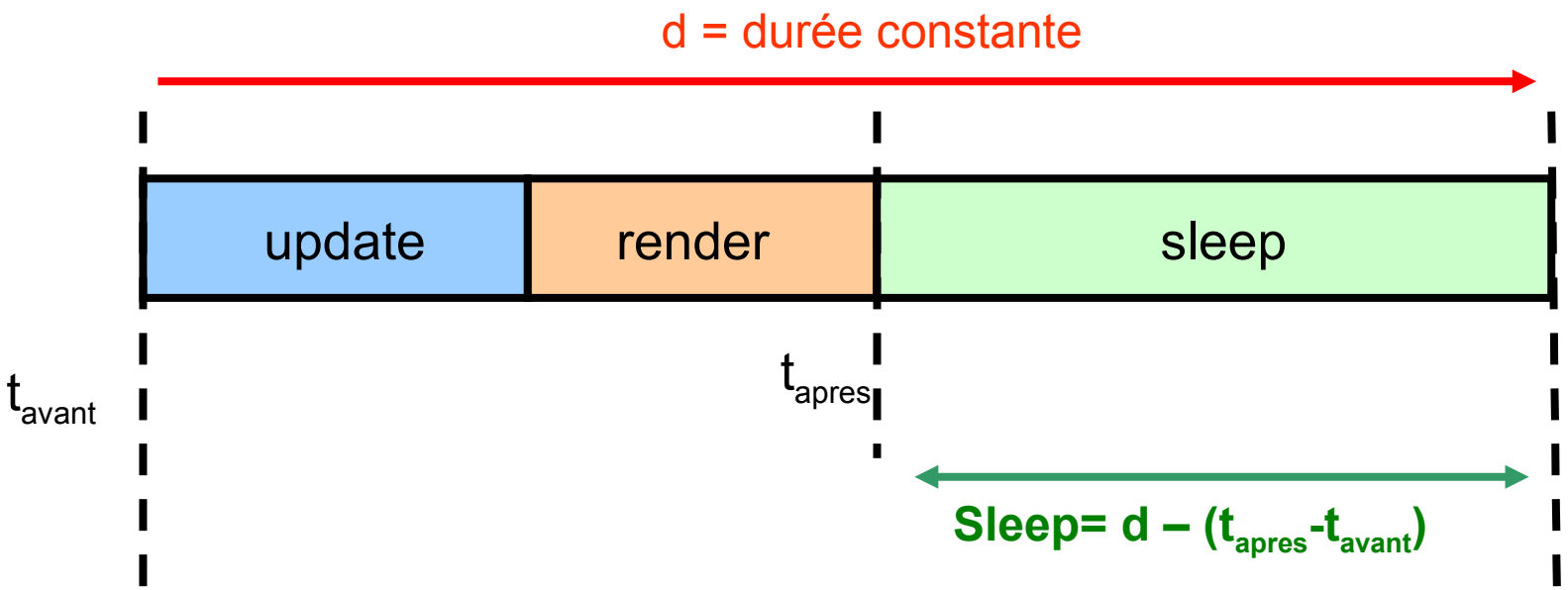
- Temps d'attente constant





Moteur de jeu

- Gestion du temps
  - Mise à jour reguliere

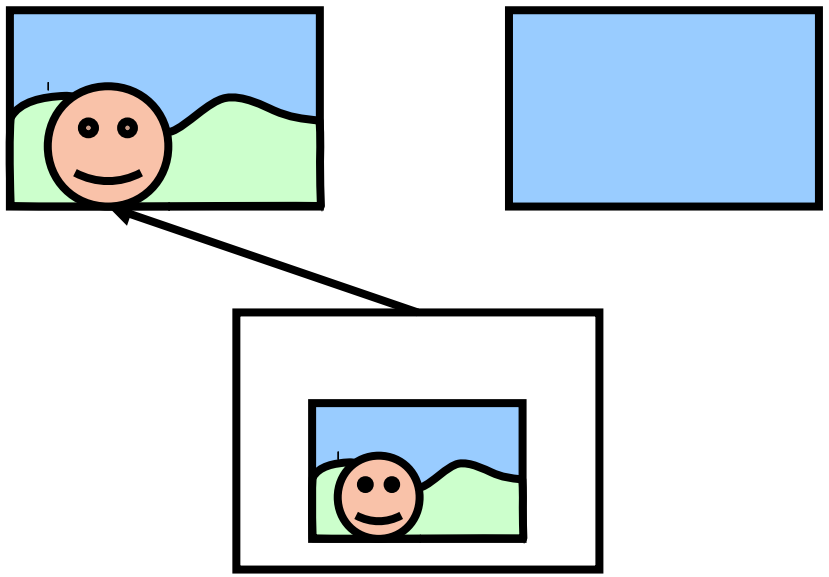


  - Sous pygame

clock.tick(60)

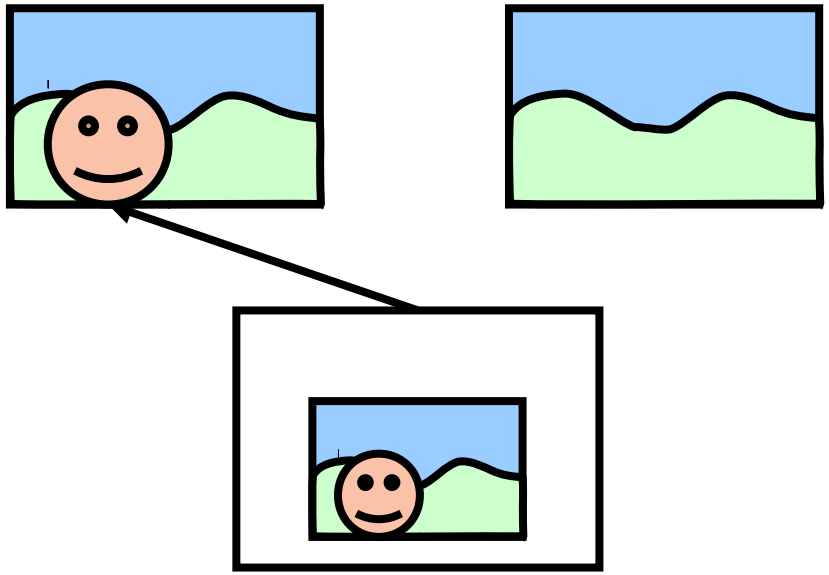
Moteur de jeu

- Double buffering
  - Une image affichée, une image cachée



Moteur de jeu

- Double buffering
  - Une image affichée, une image cachée

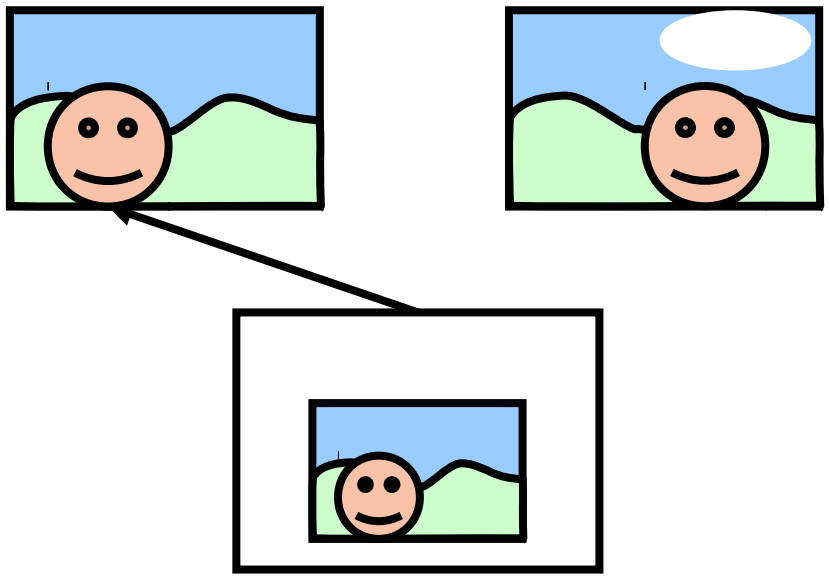


Méthode de dessin Pygame

pygame.draw.rect()

Moteur de jeu

- Double buffering
  - Une image affichée, une image cachée

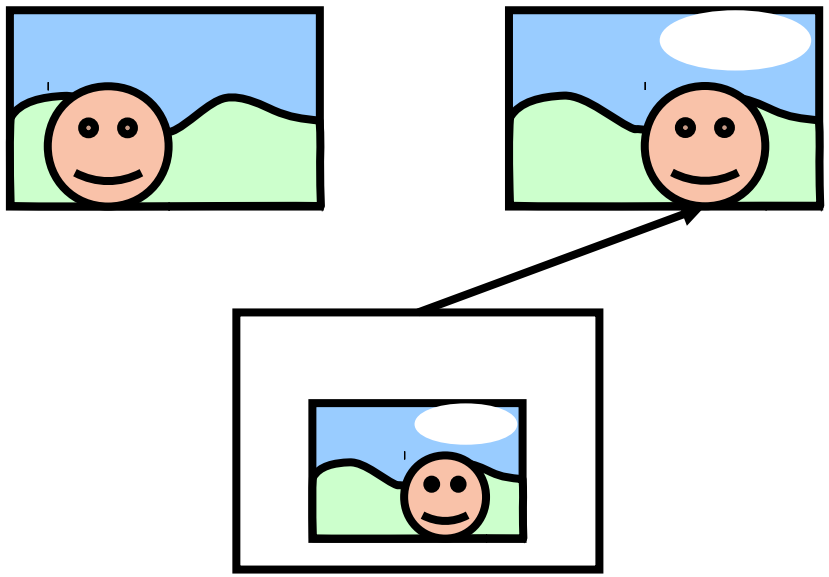


Méthode de dessin Pygame

pygame.draw.rect()

Moteur de jeu

- Double buffering
  - Une image affichée, une image cachée
  - On "inverse" les images



Inversion écrans Pygame

pygame.display.flip()

Moteur de jeu

- Gestion des evenements

- Pygame cache aspect evenementiel
  - Evenements dans une liste (pygame.event.get())

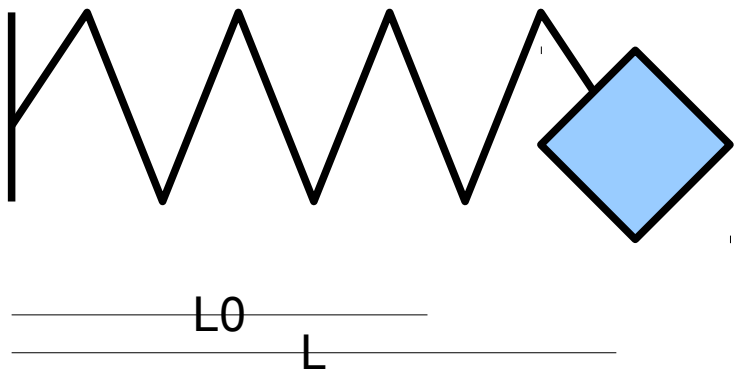
```
# --- on traite les evenements
for event in pygame.event.get():

        #si l'utilisateur arrete
        if event.type == pygame.QUIT:
                [...]

        #si l'utilisateur appuie
        if event.type == pygame.KEYDOWN:
                if event.key == pygame.K_UP:
                        [...]
```

Plan

- Mecanique du point
- Moteur de jeu
- Mecanique du point (bis)
- Comportements collectifs
- Steering behaviour
- Probleme de controle

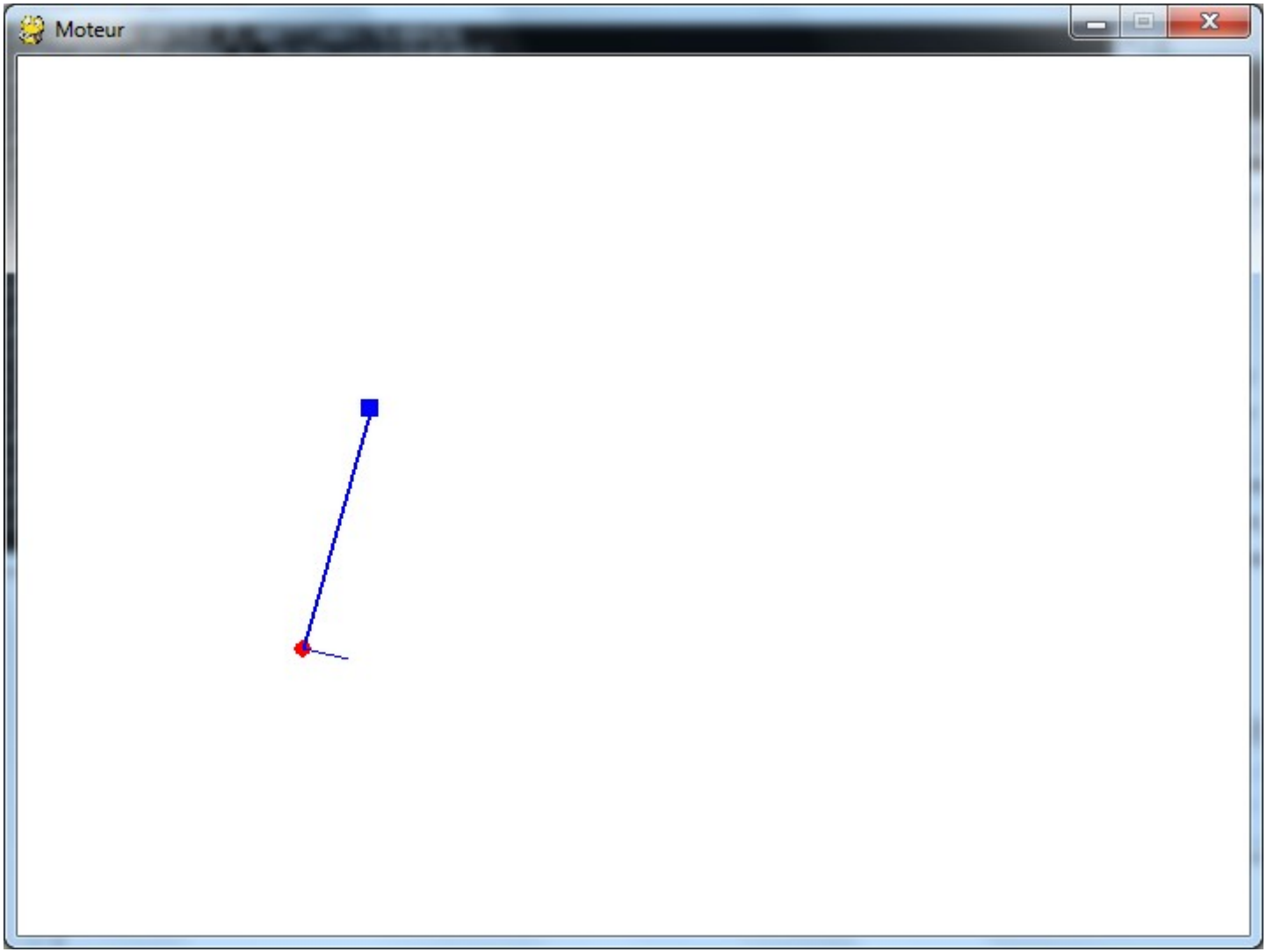
Physique du solide

- Modeles
  - Gravité constante
  - Systeme masse-ressort (raideur k)



  - Frottements
    - (- alpha.v)

Ex python

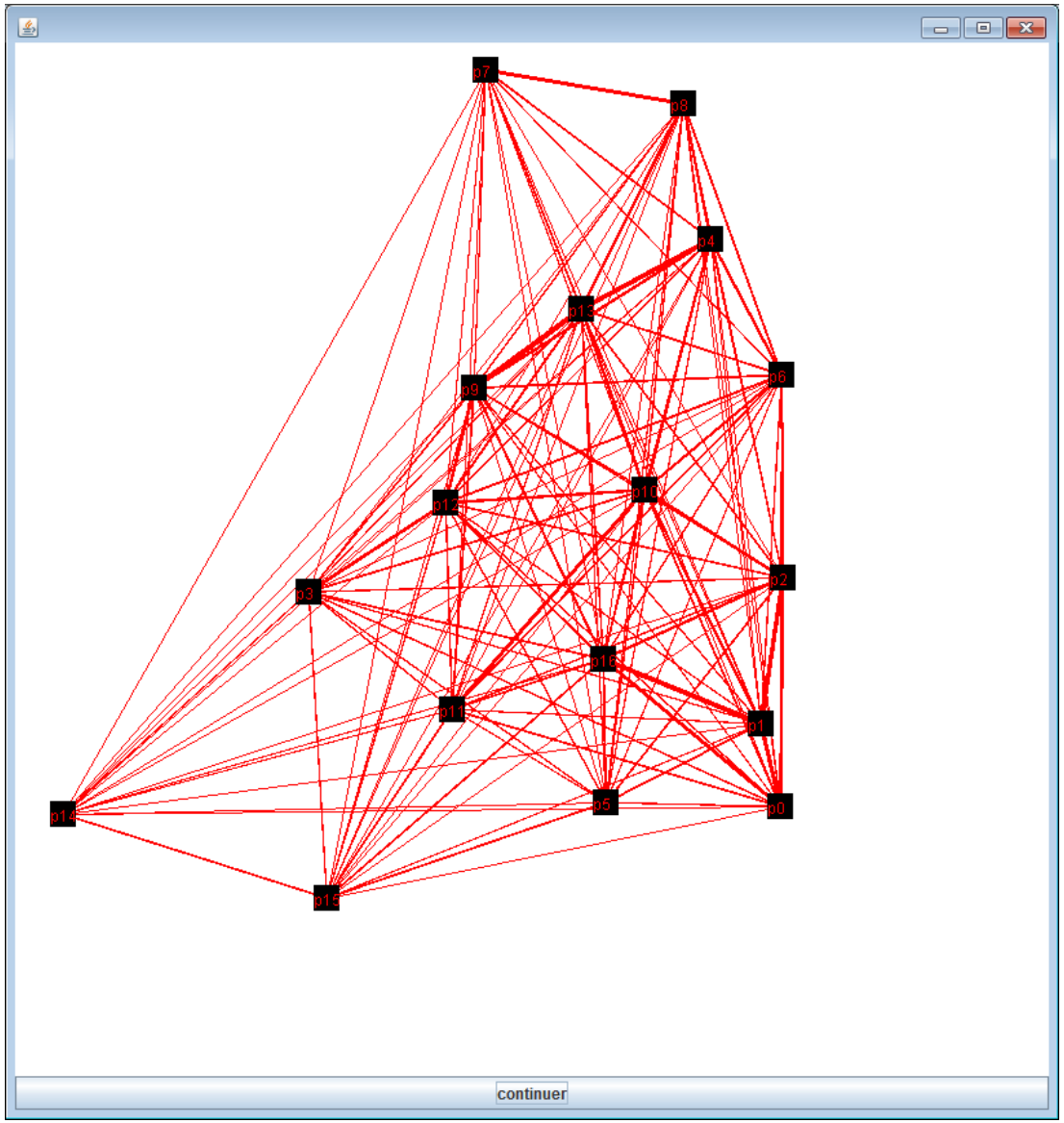


Physique du solide

- Exemples
  - Dessin de graphes
  - Treillis de ressort
- Travail étudiant
  - Dessiner espace des phases
  - Afficher la carte des forces
- Possible de ne pas supprimer affichage entre frame

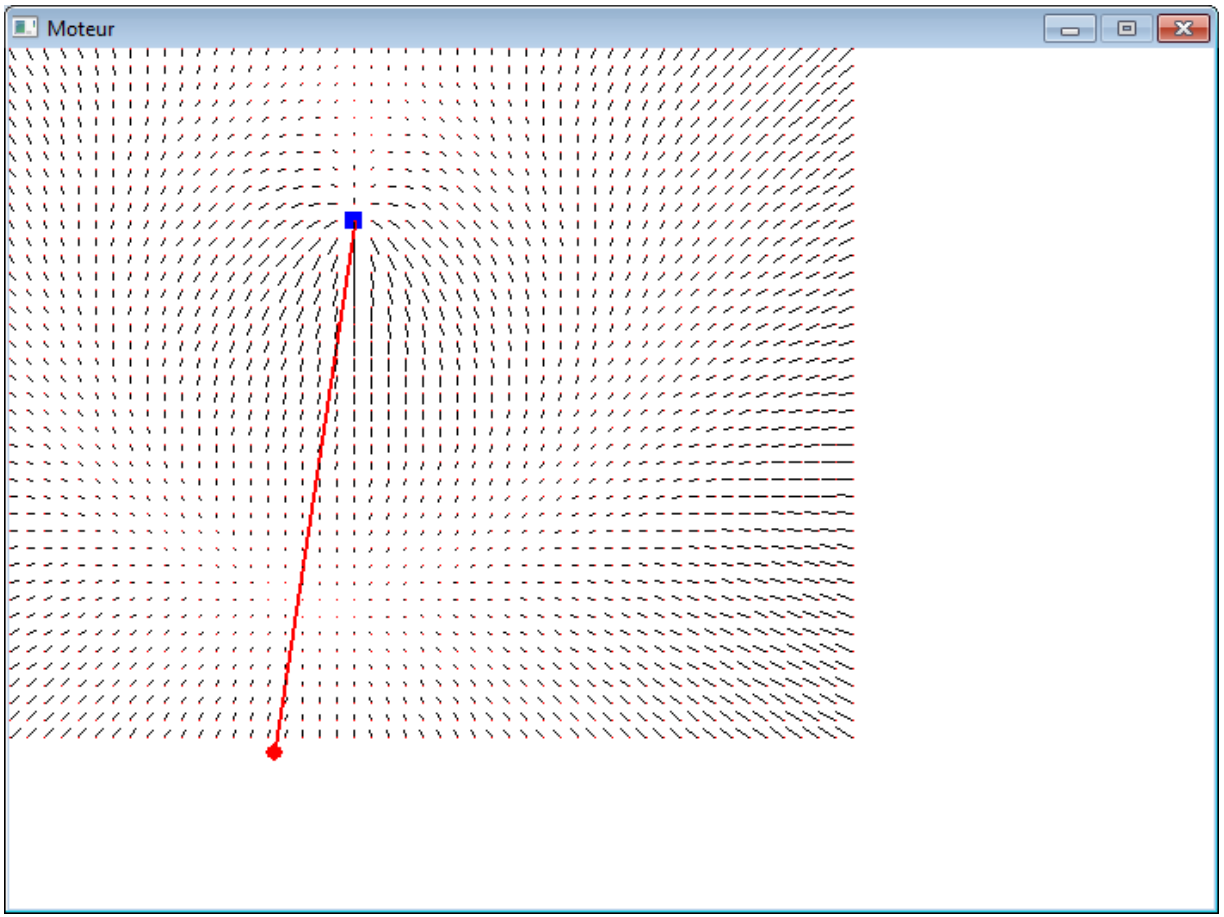
Treillis de ressort et graphe

- Démo



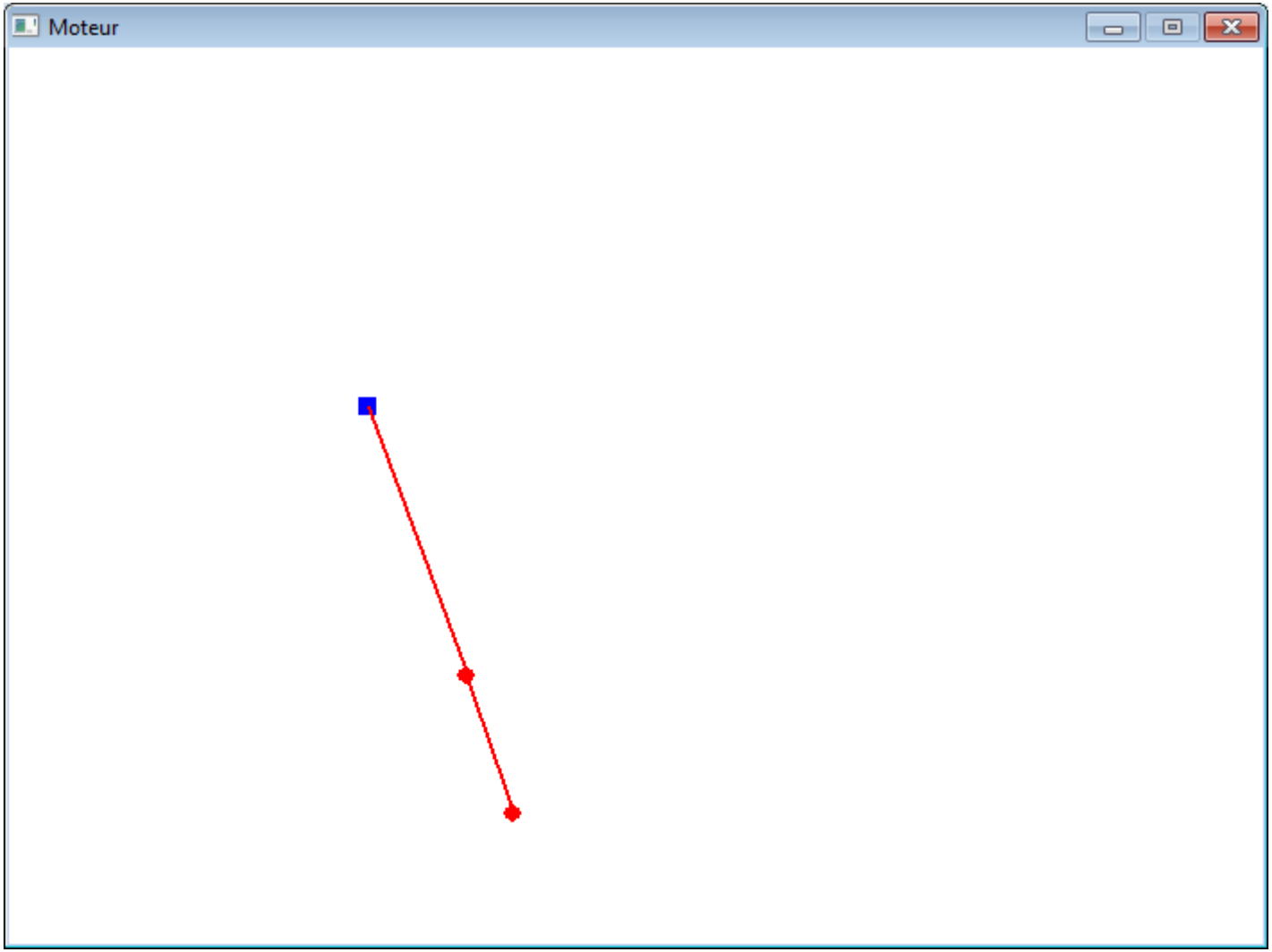
Affichage possible

- Champ de force
  - Équilibre stables et instables



exemple

- Multi-ressort

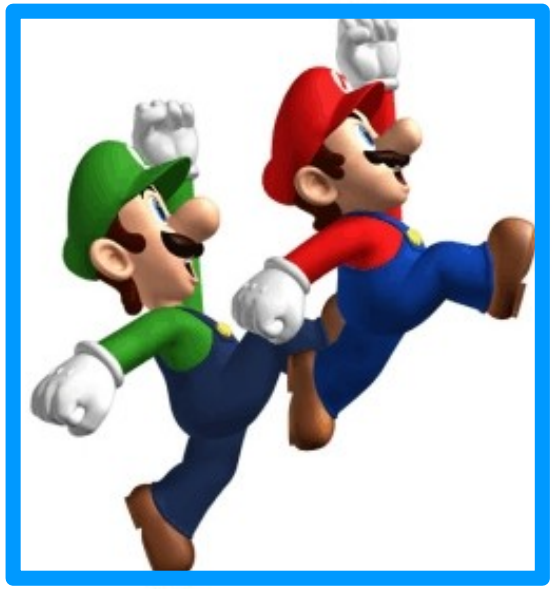


Gestionnaire de collision

- Principes
  - Utilisation de bounding box

Gestionnaire de collision

- Principes
  - Utilisation de bounding box

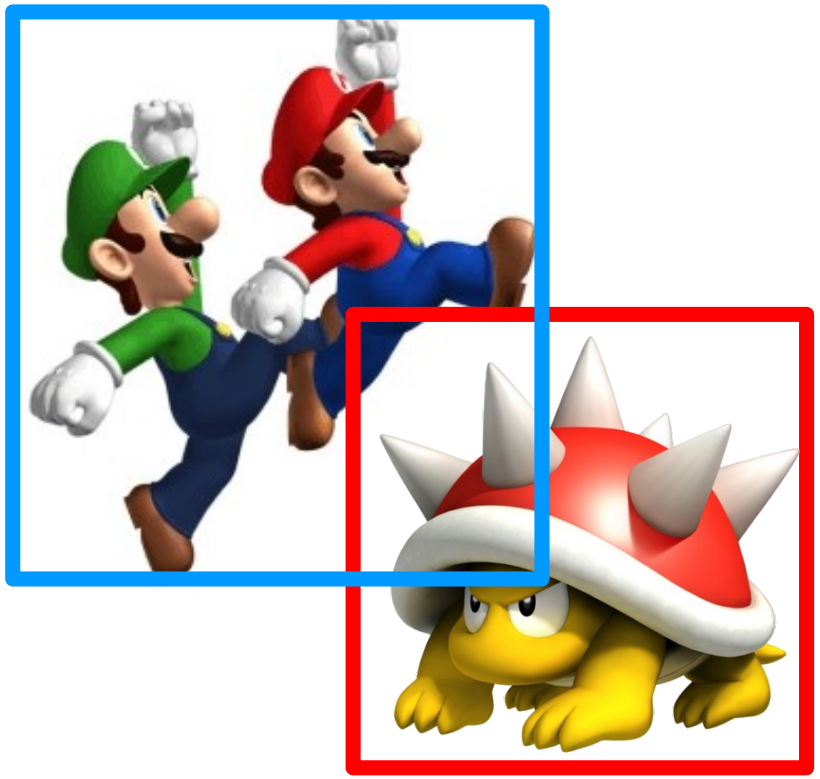


Gestionnaire de collision

- Principes
  - Utilisation de bounding box

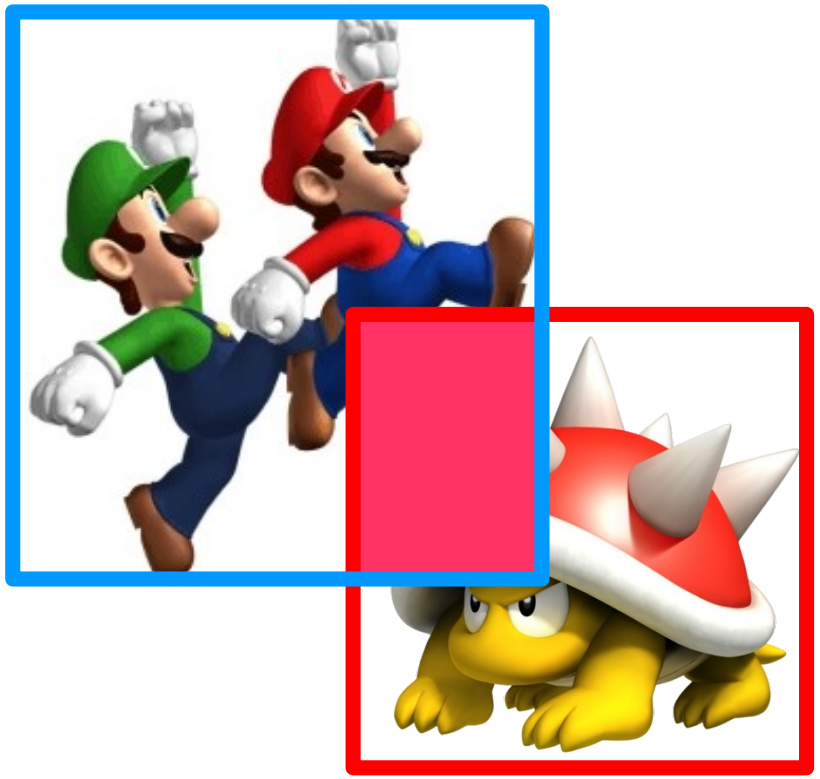
Gestionnaire de collision

- Principes
  - Utilisation de bounding box



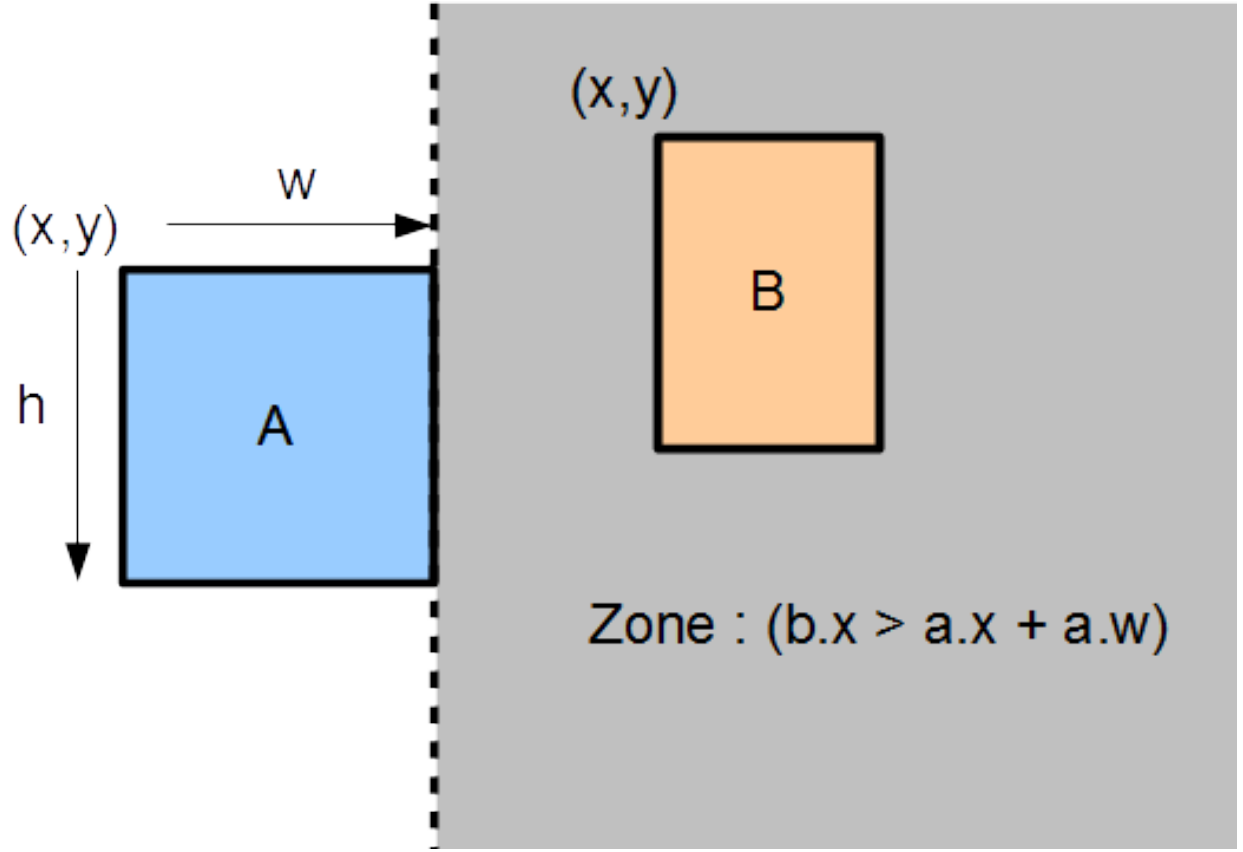
Gestionnaire de collision

- Principes
  - Utilisation de bounding box



Physique du solide

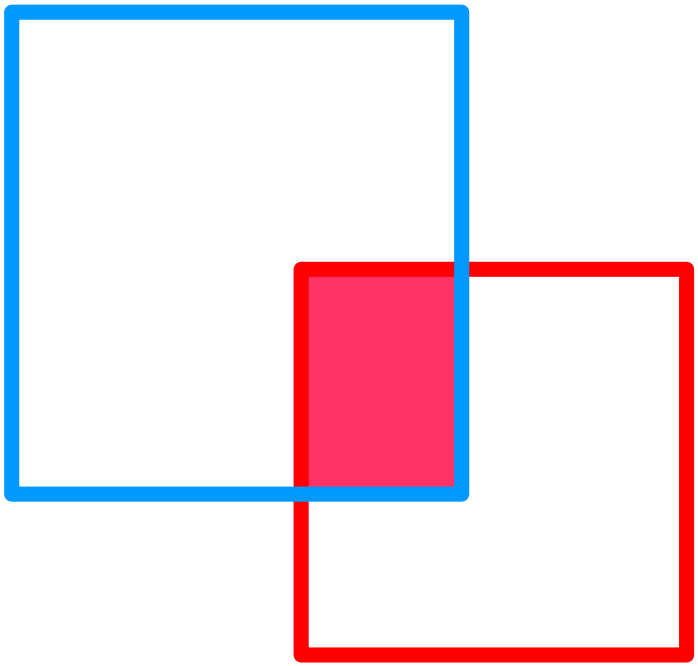
- Gestion des collisions



Gestionnaire de collision

- Intersection de rectangles

```
public boolean collide(Rect a, Rect b)
{
 If (       (a.x >= b.x + b.w)
            || (a.x + a.w <= b.x)
            || (a.y >= b.y + b.h)
            || (a.y + a.h <= b.y))
    {
        return false;
    }
    return true;
}
```

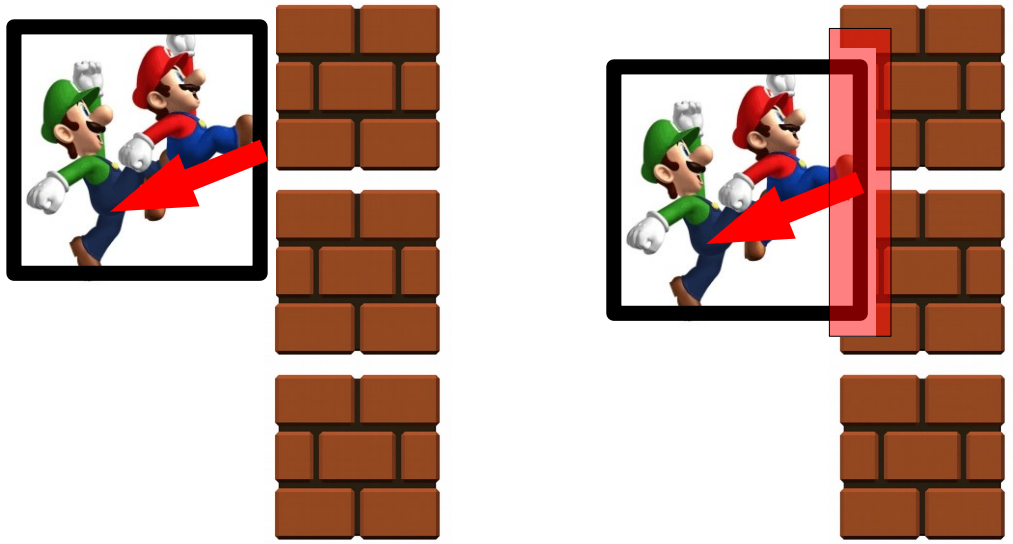


Gestionnaire de collision

- Prise en compte du résultat
  - Peu indétermination (t petit)

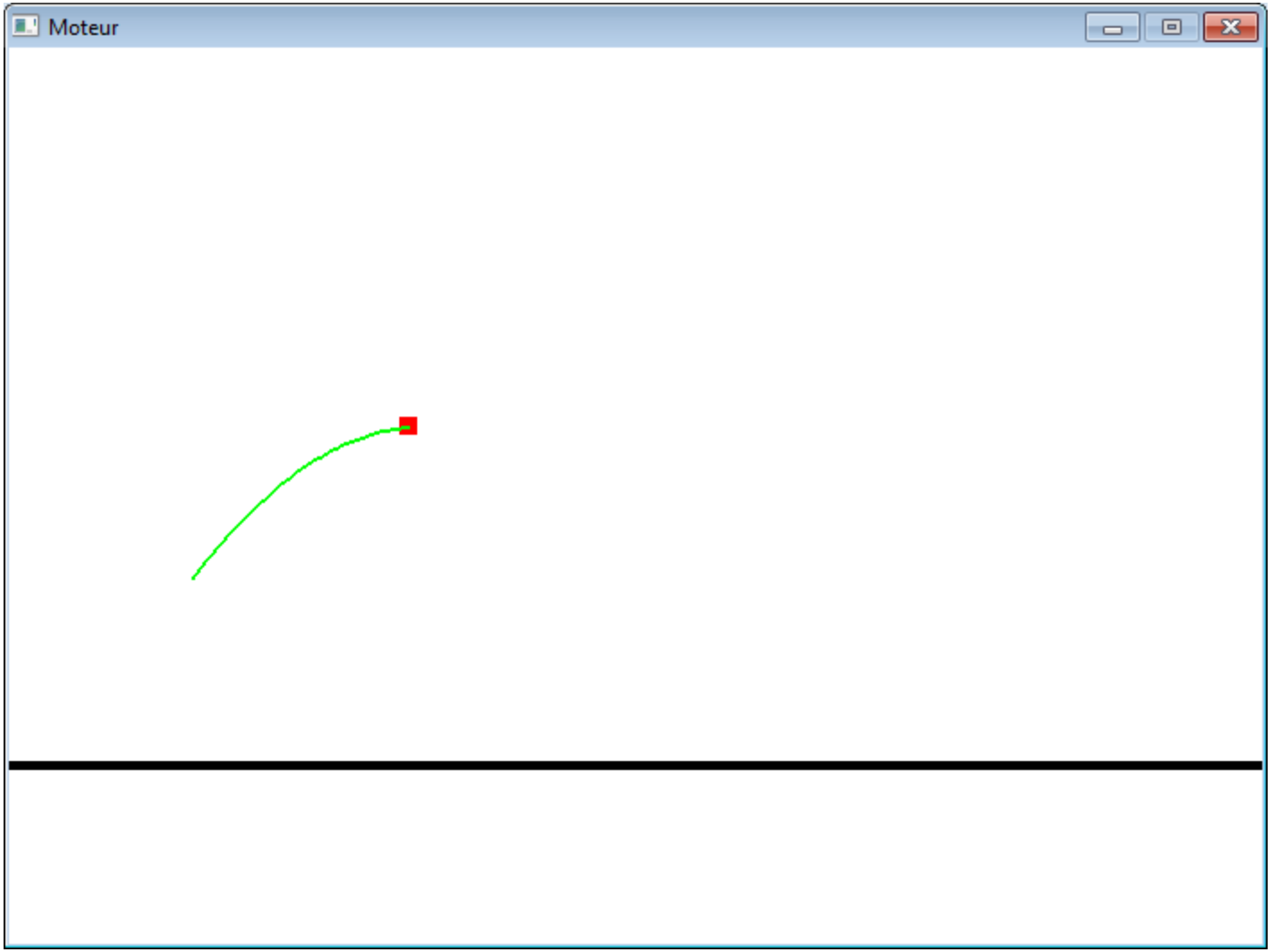
- Solution

1. Revenir passé
2. Approximation



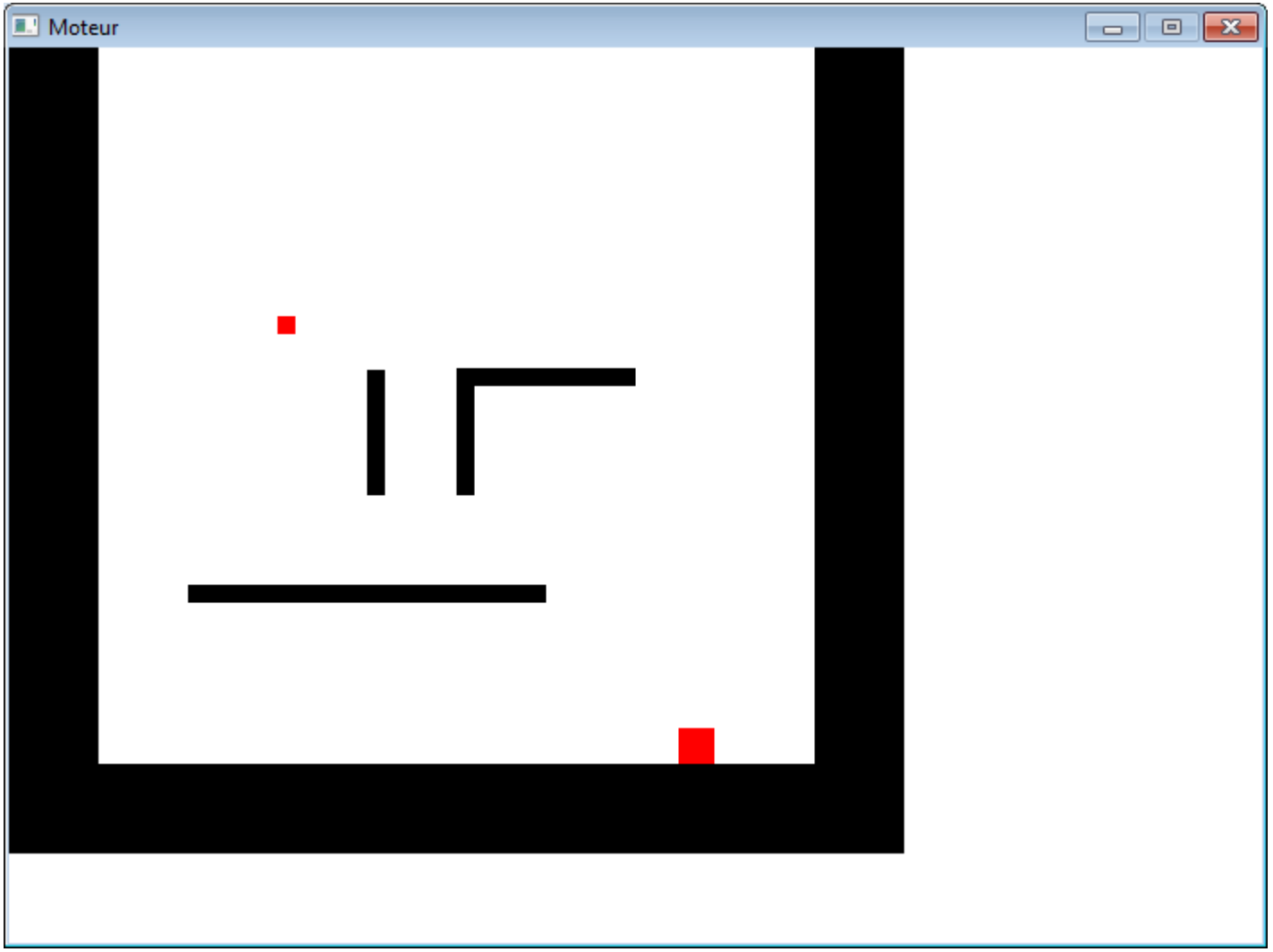
Exemple python

- Angry birds like



Exemple python

- Simulateur avec collision



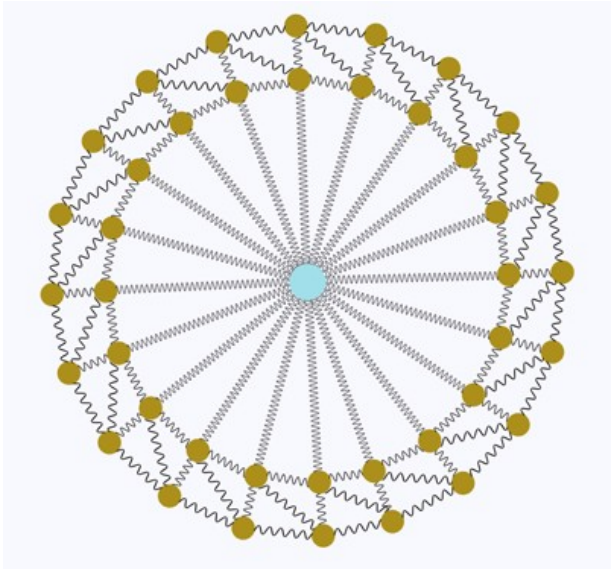
Jeu video

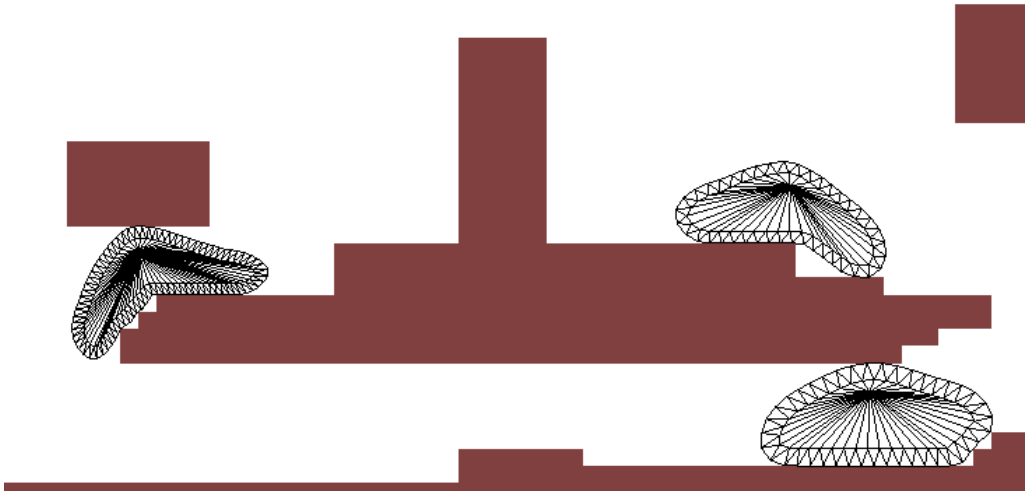
- Demonstration projet dut
  - Portal
  - Super street Melee
  - Save the train
  - Angry birds

- Autre
  - Bang
  - Gish
  - Mario-like
  - Billard (collision elastique)

Jeu video - Gish

- Système à base de ressorts





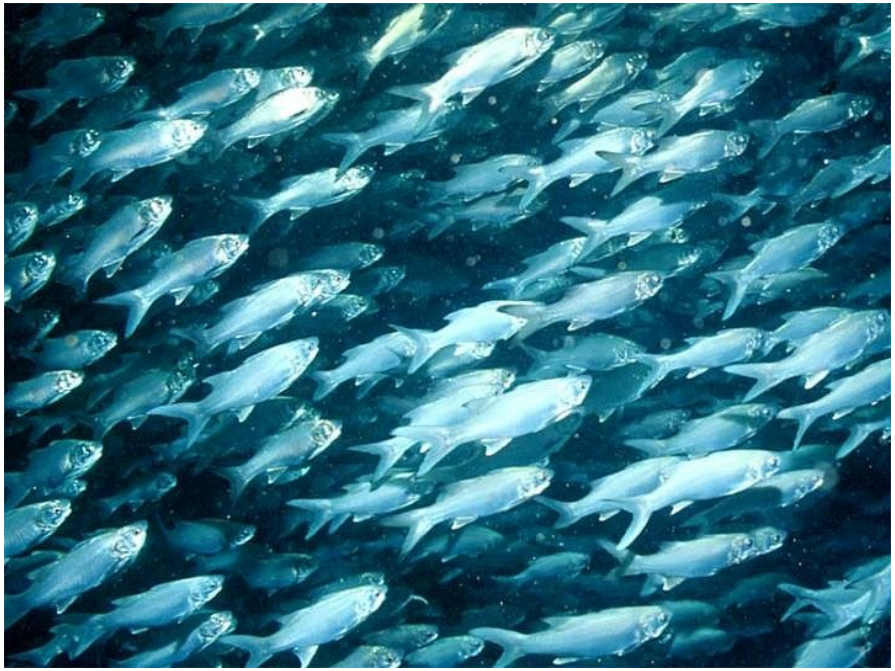
http://cowboyprogramming.com/2007/01/05/blob-physics/

Plan

- Mecanique du point
- Moteur de jeu
- Mecanique du point (bis)
- Comportements collectifs
- Steering behaviour
- Probleme de controle

Comportement

- Flocking
  - Pas de leader



Comportement

- Flocking

- Foule de piétons
  - Couche de comportement supplémentaire
  - D. Helbling "social force model" (physical rev 95)
  - Helblin "Simulating dynamical features of escape panic" (nature 2000)

Système de forces sociales

Représenter relations avec les autres par des forces (proxemie)

http://www.larecherche.fr/savoirs/cognition/comment-pietons-marchent-foule-01-03-2011-87528

Plan

- Mecanique du point
- Moteur de jeu
- Mecanique du point (bis)
- Comportements collectifs
- Steering behaviour
- Probleme de controle

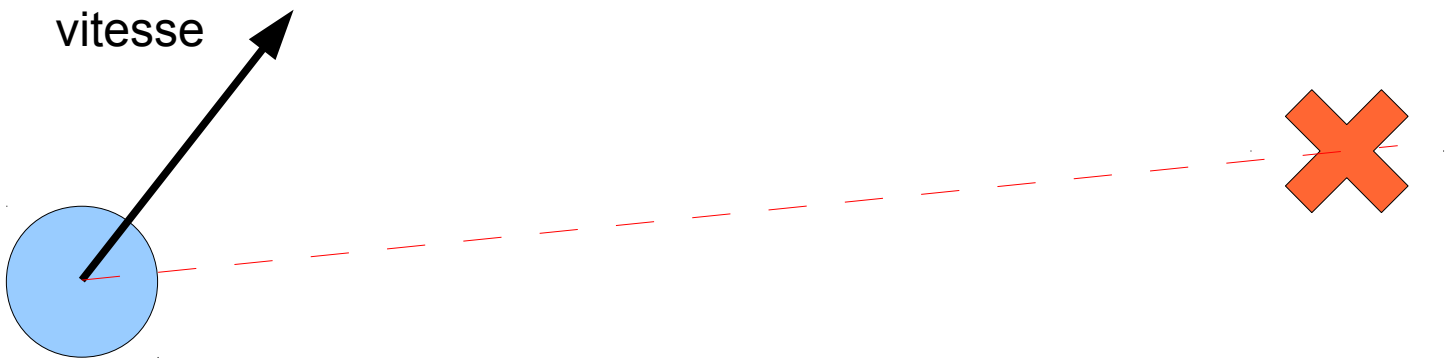
Steering behaviour

- Modeliser comportement
  - par acceleration/vitesse

- Article de référence
  - Steering Behaviors For Autonomous Characters (1999)
  - Craig Reynolds (http://www.red3d.com/cwr/)

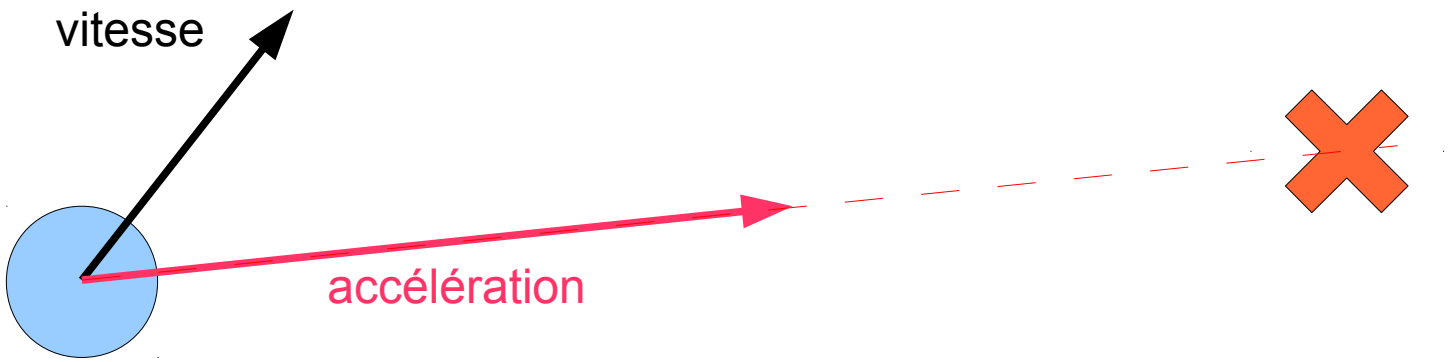
Steering beahviour

- Plein de modeles
  - Se diriger vers un point



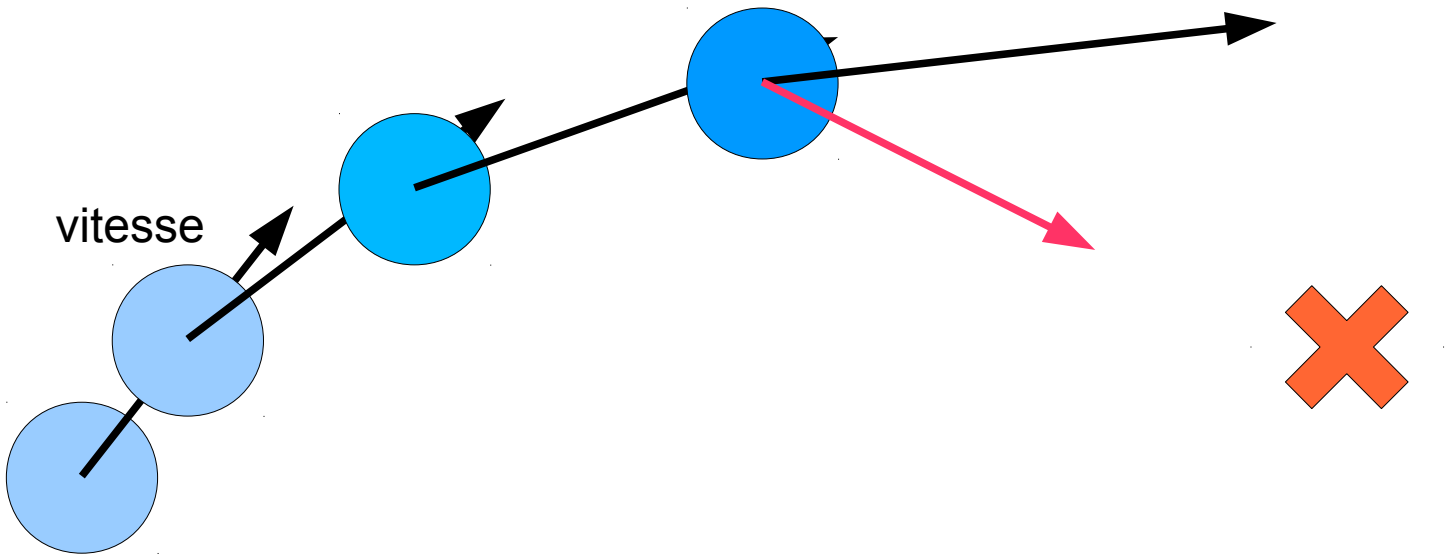
Steering beahviour

- Plein de modeles
  - Se diriger vers un point



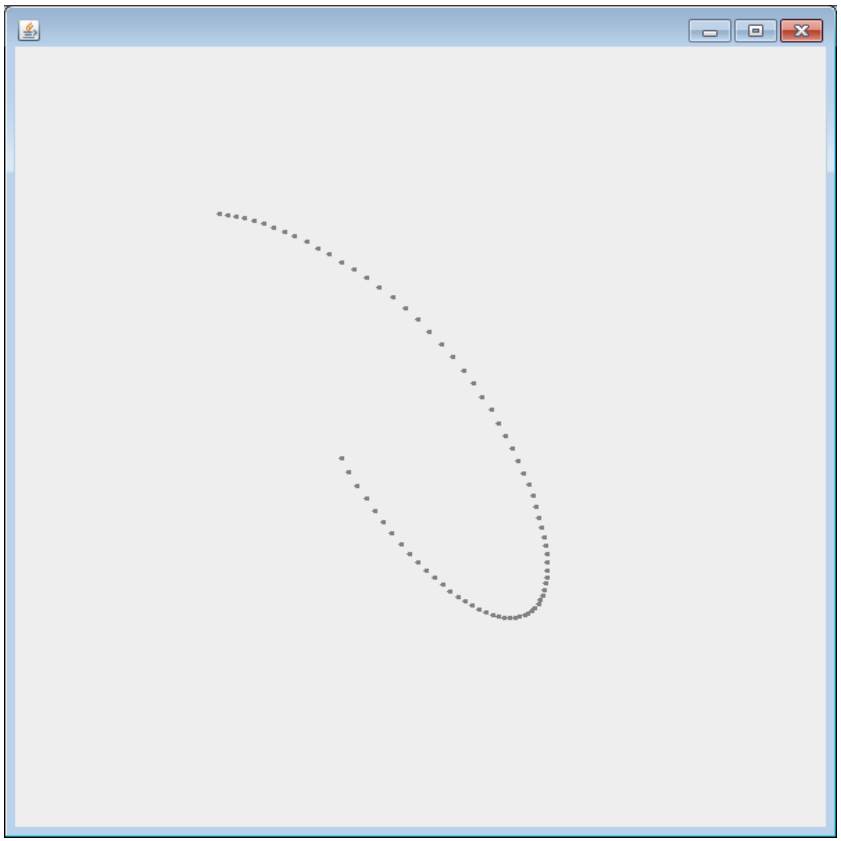
Steering beahviour

- Plein de modeles
  - Se diriger vers un point



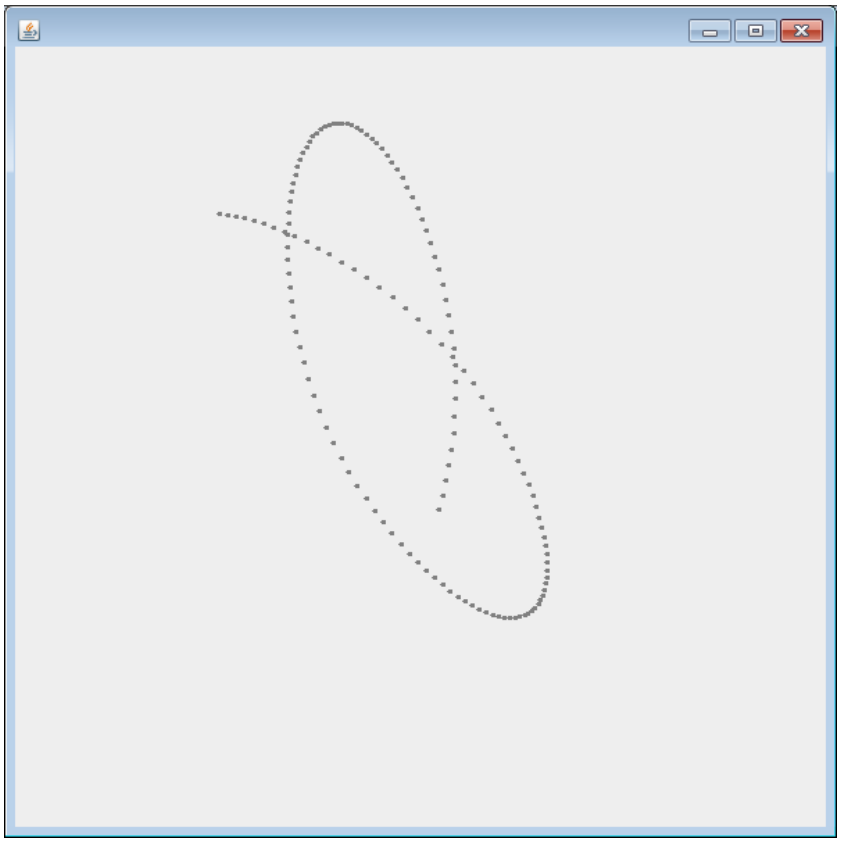
Steering beahviour

- Plein de modeles
  - Se diriger vers un point



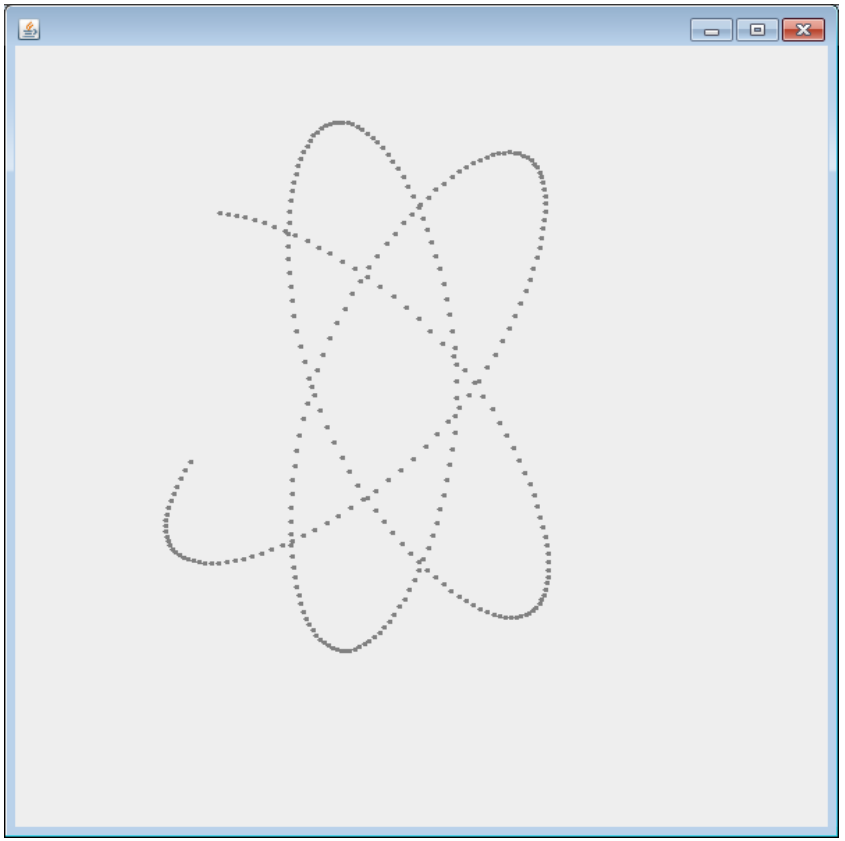
Steering beahviour

- Plein de modeles
  - Se diriger vers un point



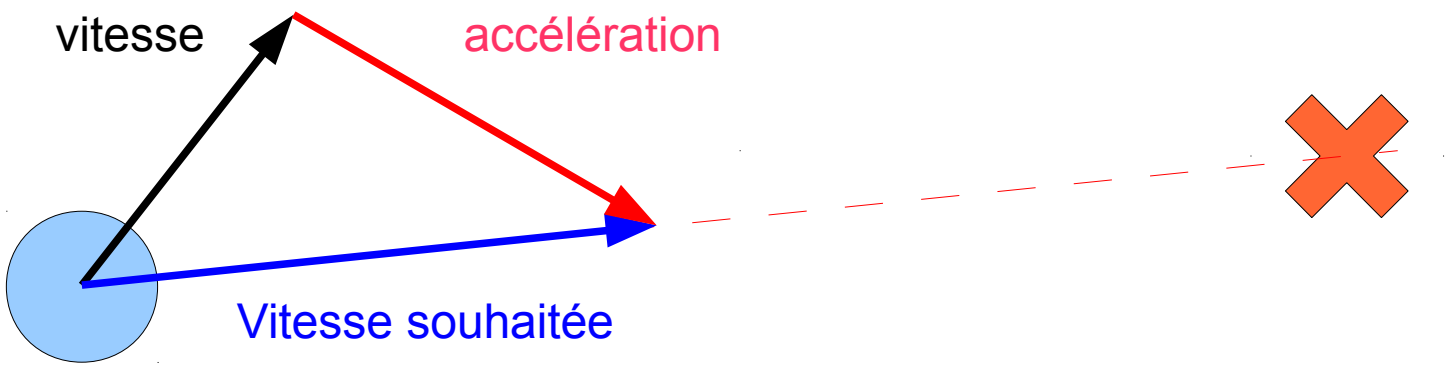
Steering beahviour

- Plein de modeles
  - Se diriger vers un point
  - Mauvaise solution : oscillateur

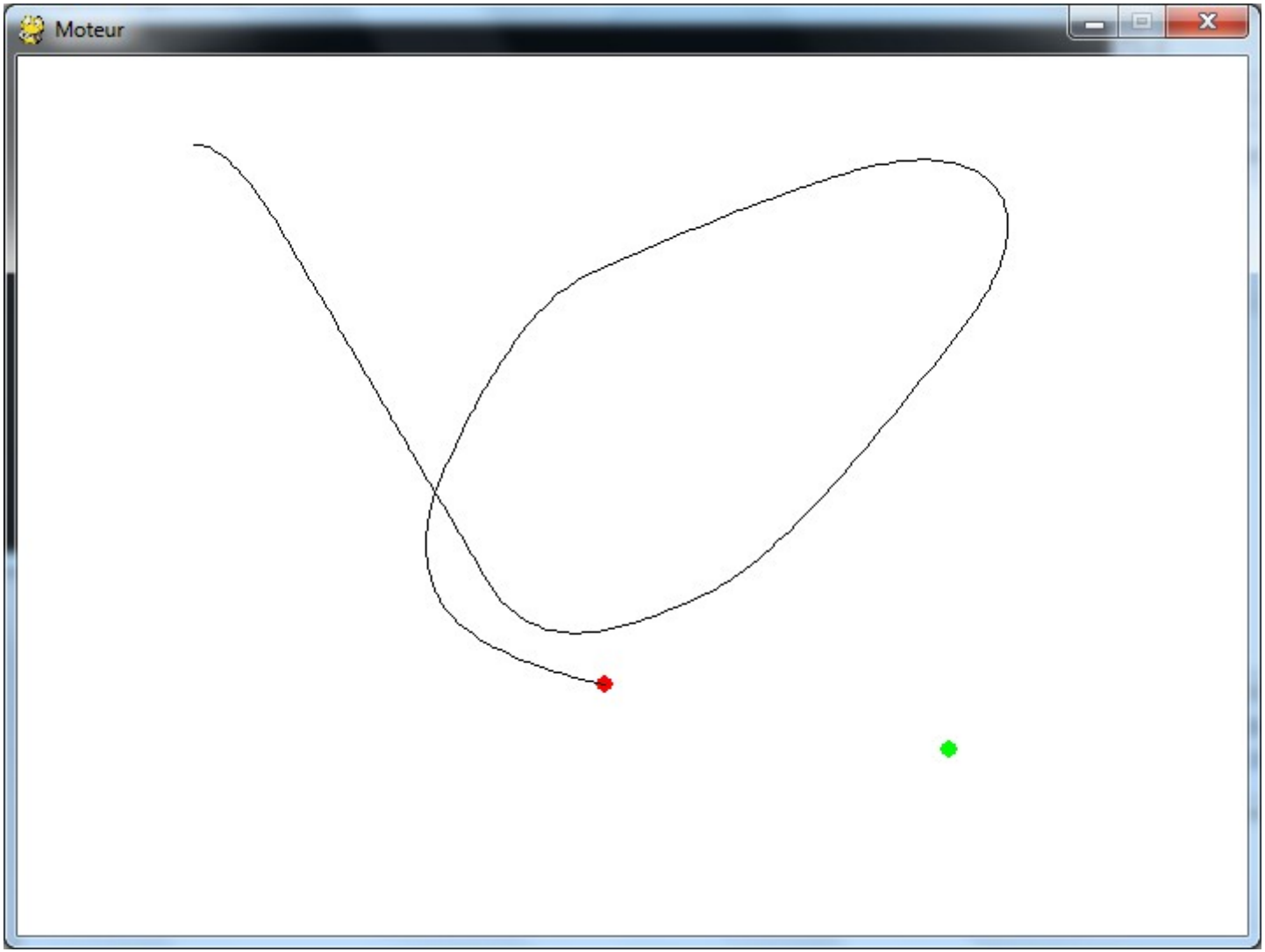


Steering beahviour

- Plein de modeles
  - Se diriger vers un point

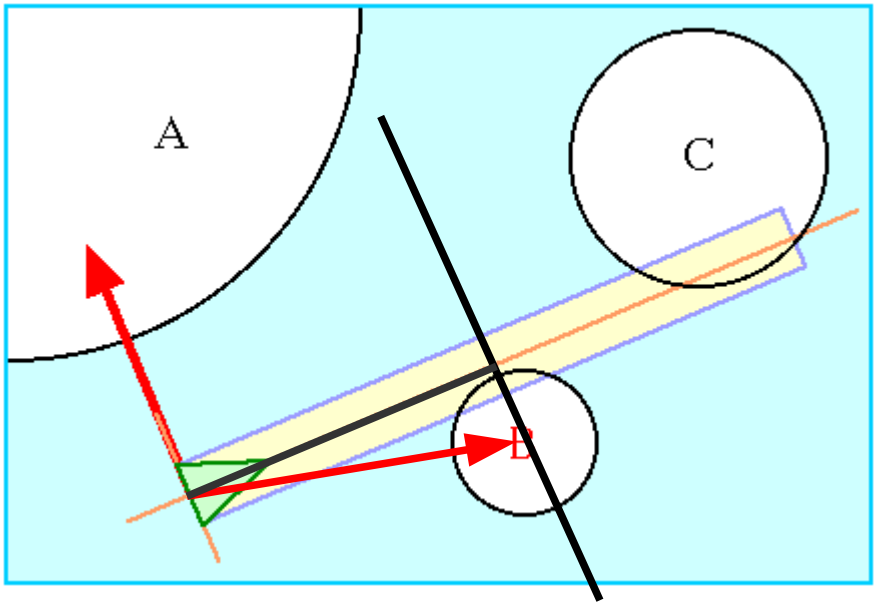


demo



Steering beahviour

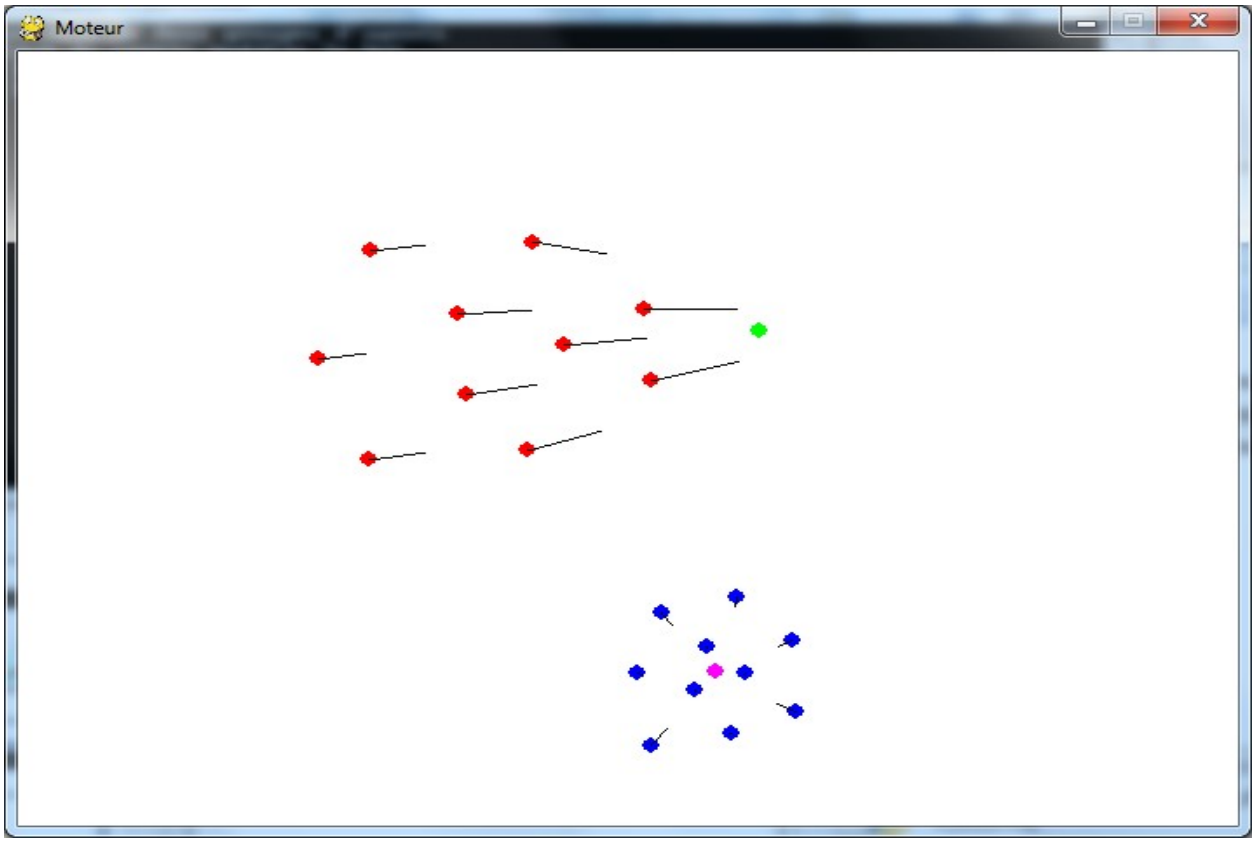
- Plein de modeles
  - Evitement



Craig Reynolds – Steering behaviours

Steering beahviour

- Plein de modeles
  - Force sociales
    - Répulsion entre agents proches



Steering beahviour

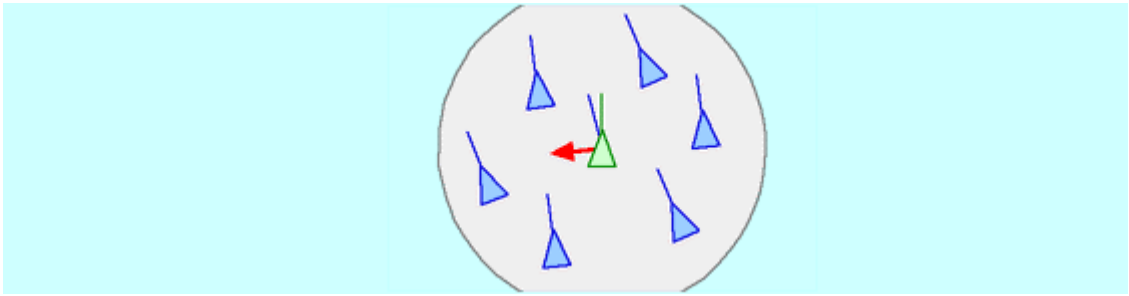
- Modèle de flocking
  - 3 comportements

http://gamedevelopment.tutsplus.com/tutorials/the-three-simple-rules-of-flocking-behaviors-alignment-cohesion-and-separation--gamedev-3444

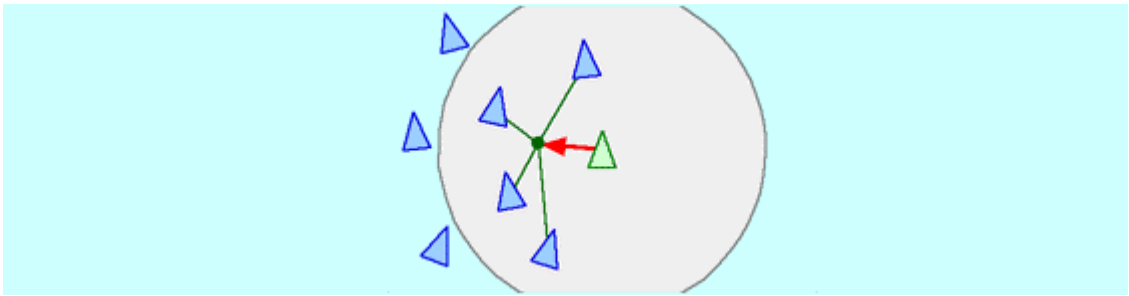
Steering beahviour

- Modèle de flocking
  - 3 comportements

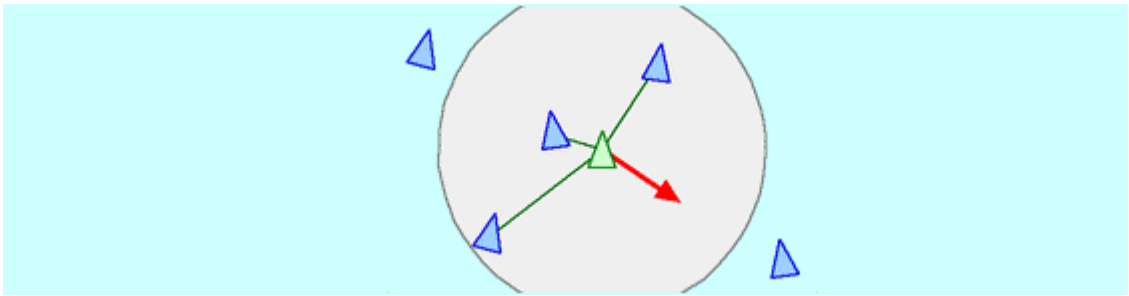
Alignement : oriente vitesse selon la moyenne des voisins



Cohésion : vitesse vers baricentre des voisins



Separation : les voisins proches repoussent l'agent



http://gamedevelopment.tutsplus.com/tutorials/the-three-simple-rules-of-flocking-behaviors-alignment-cohesion-and-separation--gamedev-3444

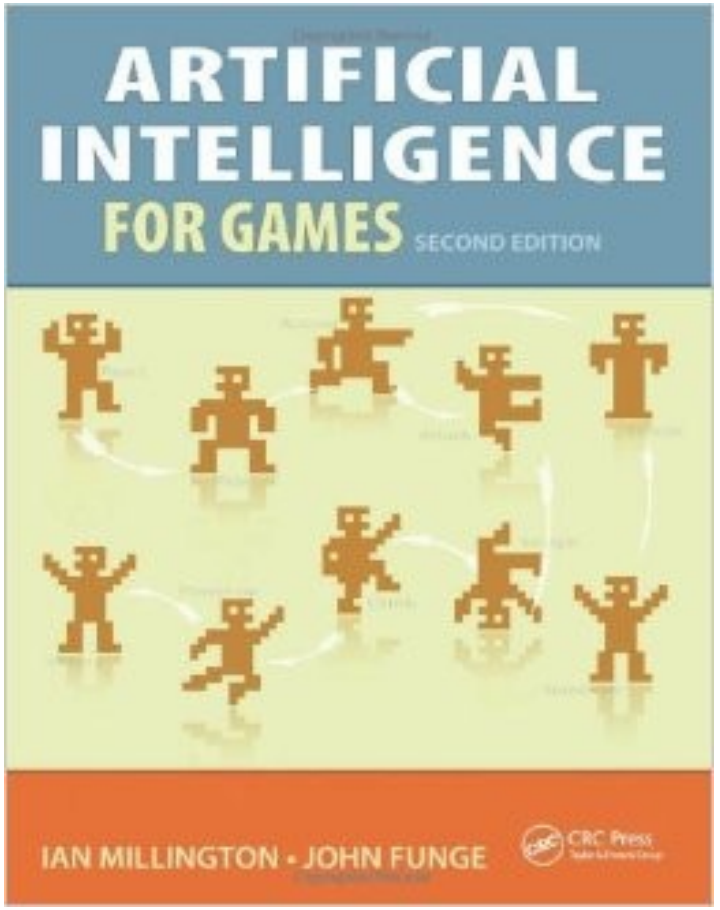
Steering behaviour

- Assassin's creed

Steering behaviour

- Ia dans les jeux vidéos
  - Millington, Funge (2006)

- http://ai4g.com/



Plan

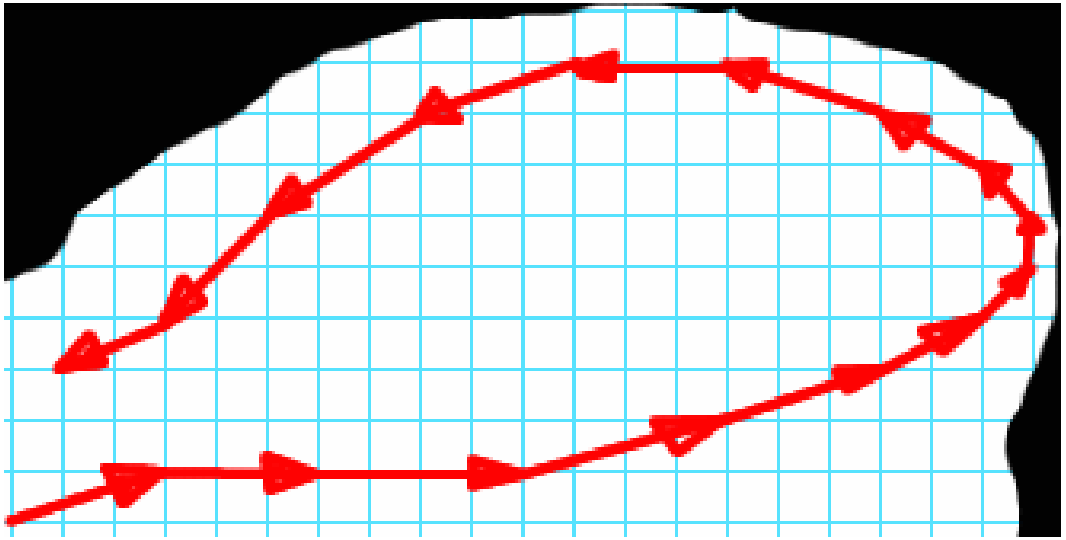
- Mecanique du point
- Moteur de jeu
- Mecanique du point (bis)
- Comportements collectifs
- Steering behaviour
- Probleme de controle

Probleme de controle

- Systeme dynamique
  - État interne du système
  - Actions possibles
- Loi d'évolution du systeme
  - $s_{t+1}=f(s_t,a_t)$
- Question
  - Quelle loi de controle a_t ?

Probleme de controle

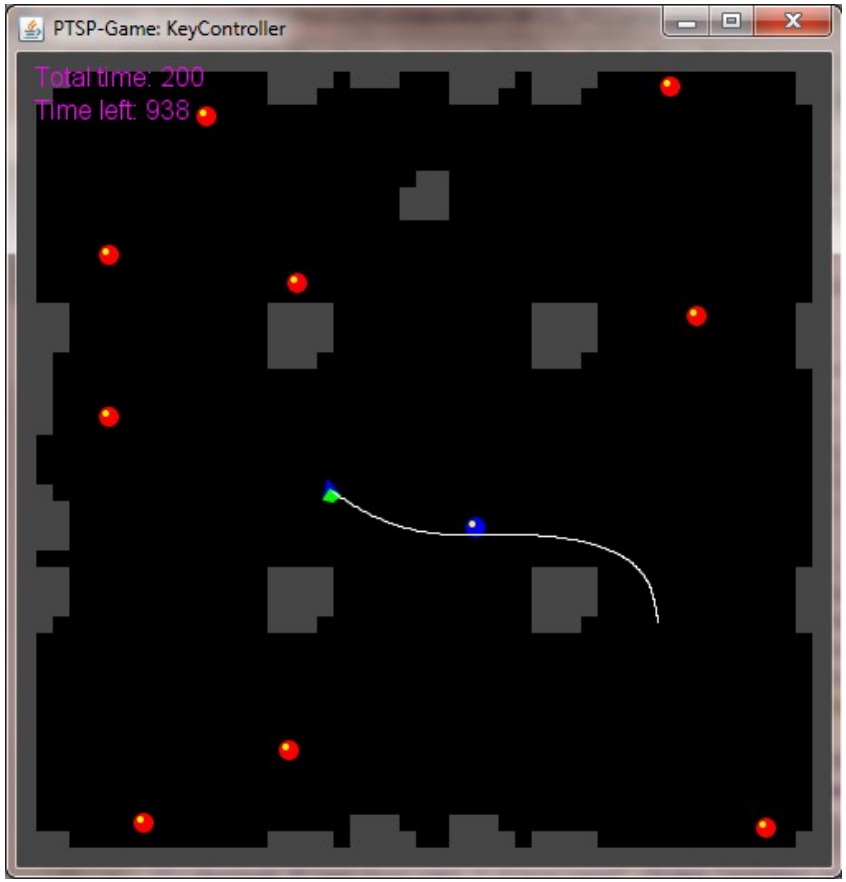
- Jeu de course papier



http://www.sjbaker.org/paper_and_pencil_games/graph_racers/

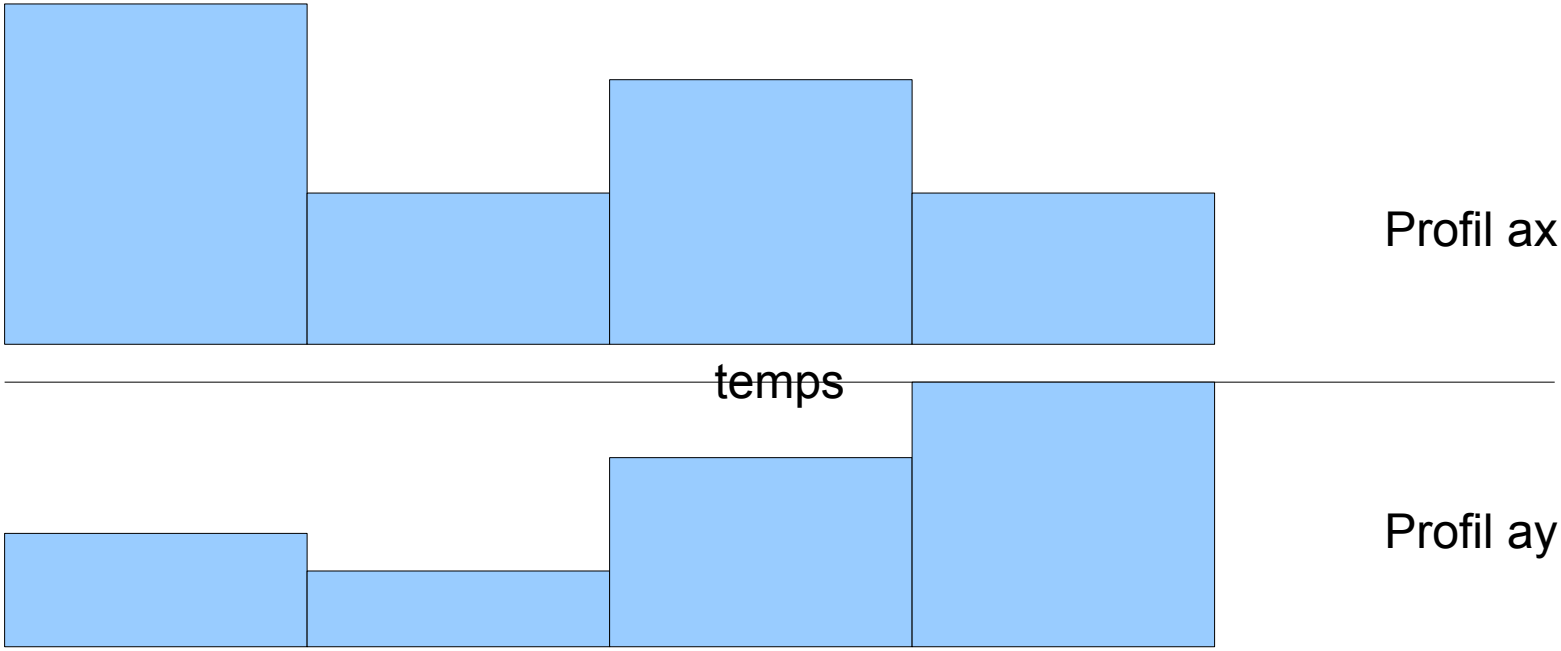
Voyageur de commerce

- Différences
  - Voyageur de commerce
  - Physical TSP (http://www.ptsp-game.net/)



Controle

- Raisonner sur des profils d'accélération



- Optimiser ces profils (recherche dans espace des fonctions)
  - Parcours aléatoire
  - Descente de gradient sur parametres
  - Algorithme genetiques

Controle

- Karl Sims
  - Evolved virtual creature (Siggraph '94)
  - http://www.karlsims.com/evolved-virtual-creatures.html

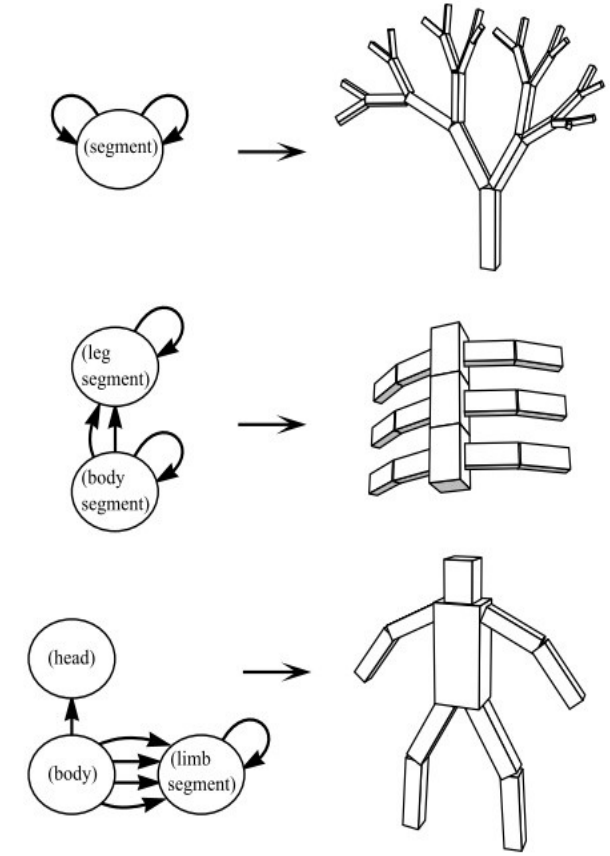


Figure 1: Designed examples of genotype graphs and corresponding creature morphologies.

Karl Sims

- Évolution forme + controle
  - Meilleures formes pour résoudre une tâche

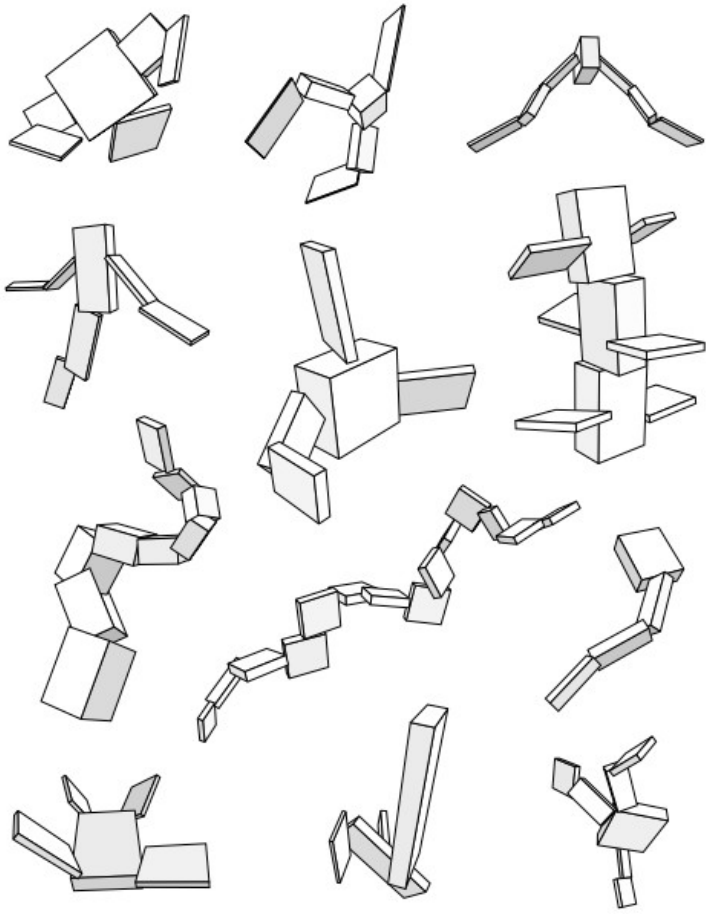


Figure 6: Creatures evolved for swimming.

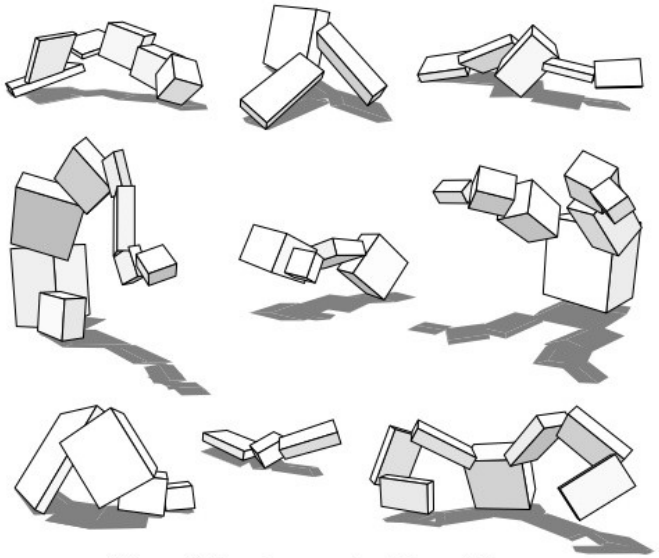


Figure 7: Creatures evolved for walking.

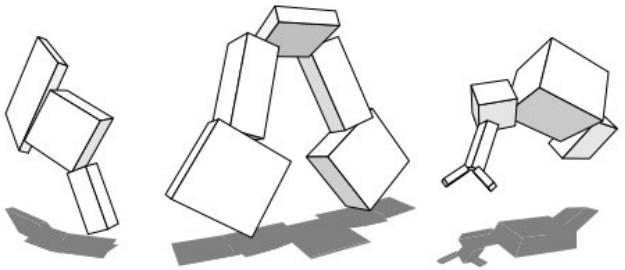
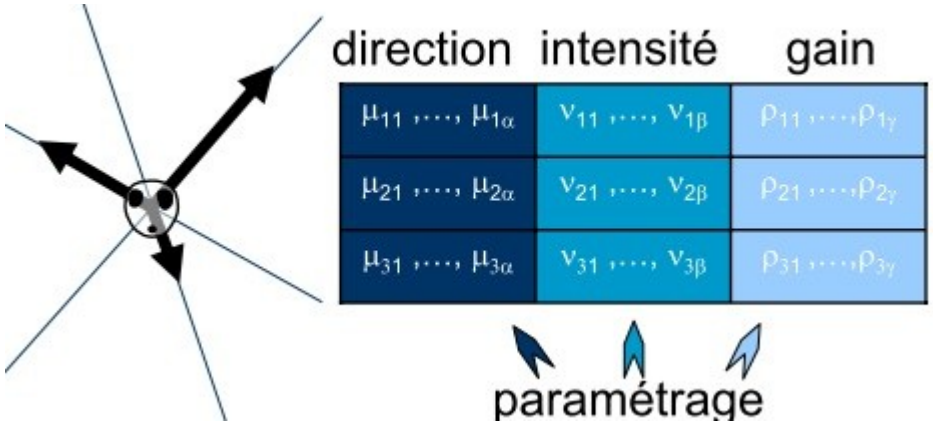
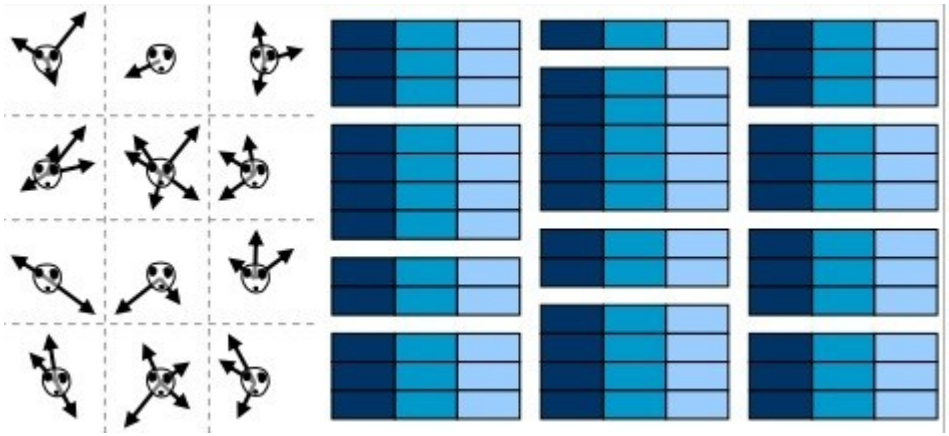


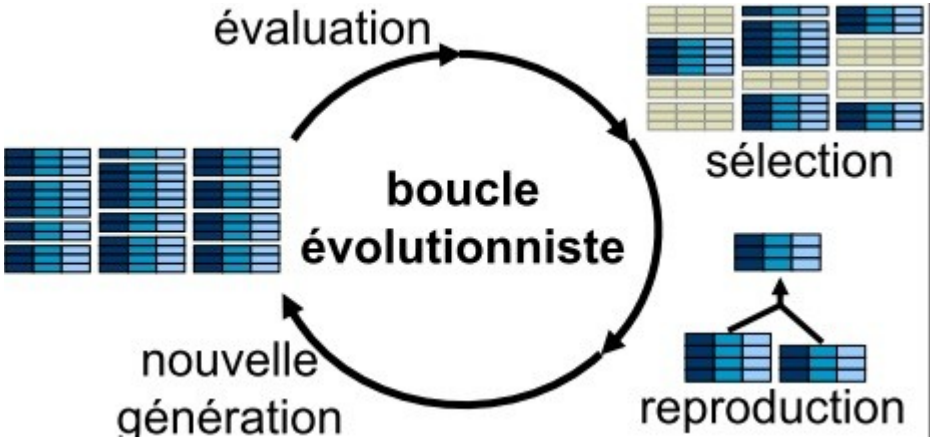
Figure 8: Creatures evolved for jumping.

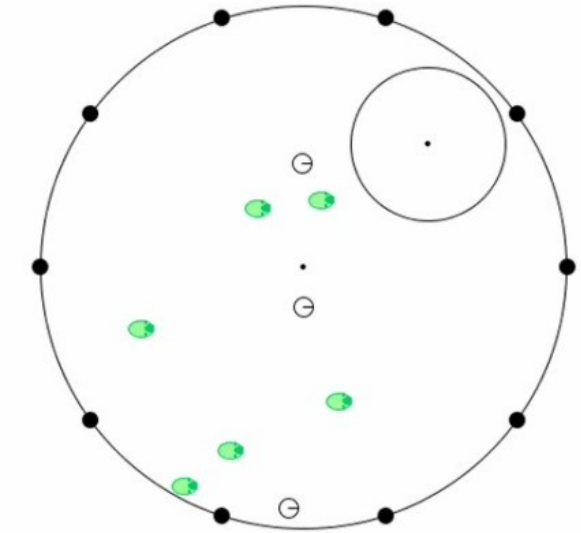
Gacs - Lois de controle

- GACS (Flacher 2005)









Probleme : capturer des moutons