

Spel-AI

**AI, fysik och animation
Tillståndsmaskiner
Världsrepresentation
Metriker
Influence maps
Flocking**

Ingemar
Ragnemalm
ingis@isy.liu.se

Spel-AI och spel-fysik

Tillämpningar av fältens grundläggande tekniker.

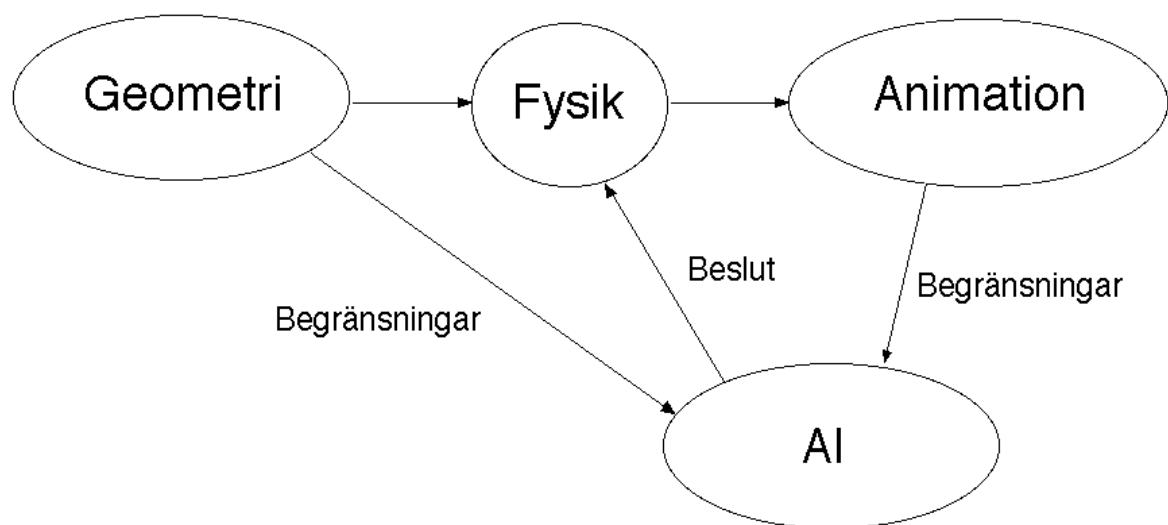
AI: Planering, beteenden, beslutsfattande

Fysik: Snarast mekanik och en del klassisk fysik

Animationer är i hög grad resultat av dessa.

Ingemar
Ragnemalm
ingis@isy.liu.se

AI, fysik och animation i spel



Ingemar
Ragnemalm
ingis@isy.liu.se

Spel-AI

Äntligen en “artificiell intelligens” som vanligt folk begriper! Ett genombrott för hela AI-begreppet?

Ingemar
Ragnemalm
ingis@isy.liu.se

Typiska spel-AI-problem

- Planering (vägsökning)
 - Beteenden
 - Beslutsfattande

Ingemar
Ragnemalm
ingis@isy.liu.se

Två sorters spel-AI

- Agenter: fiender och andra datorstyrda spelobjekt
- Abstrakta kontrollsysten, övergripande beteenden hos t.ex. en armé eller en skock får

Ingemar
Ragnemalm
ingis@isy.liu.se

Viktiga problem att ta hänsyn till:

Spatiell information

Världens geometri, andra agenter, spelaren, tittriktningen

Minne

Hur lagrar vi det agenten skall komma ihåg?

Analys

Ofta släende enkla metoder!

Aktion

Vad skall agenten utföra? Förflyttningar mm.

Fusk

Bör agenten tillåtas fuska?

Ingemar
Ragnemalm
ingis@isy.liu.se

Vanliga metoder

- Finita tillståndsmaskiner
 - Regelsystem
 - Planering och problemlösning
- Biologi-inspirerad AI, parameteruppsättningar

Ingemar
Ragnemalm
ingis@isy.liu.se

Typisk actionspels-AI

Vanliga beteenden:

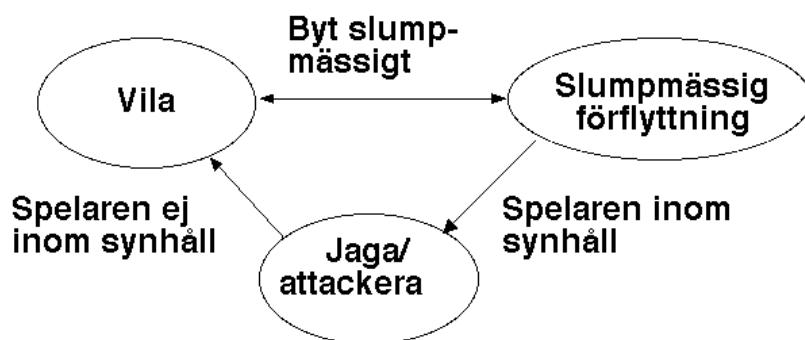
- Jaga
- Fly
- Slumpmässig förflyttning
- Följ en bana
- Vila

Ingemar
Ragnemalm
ingis@isy.liu.se

Agenten byter mellan dessa beteenden
beroende på spelhändelser:

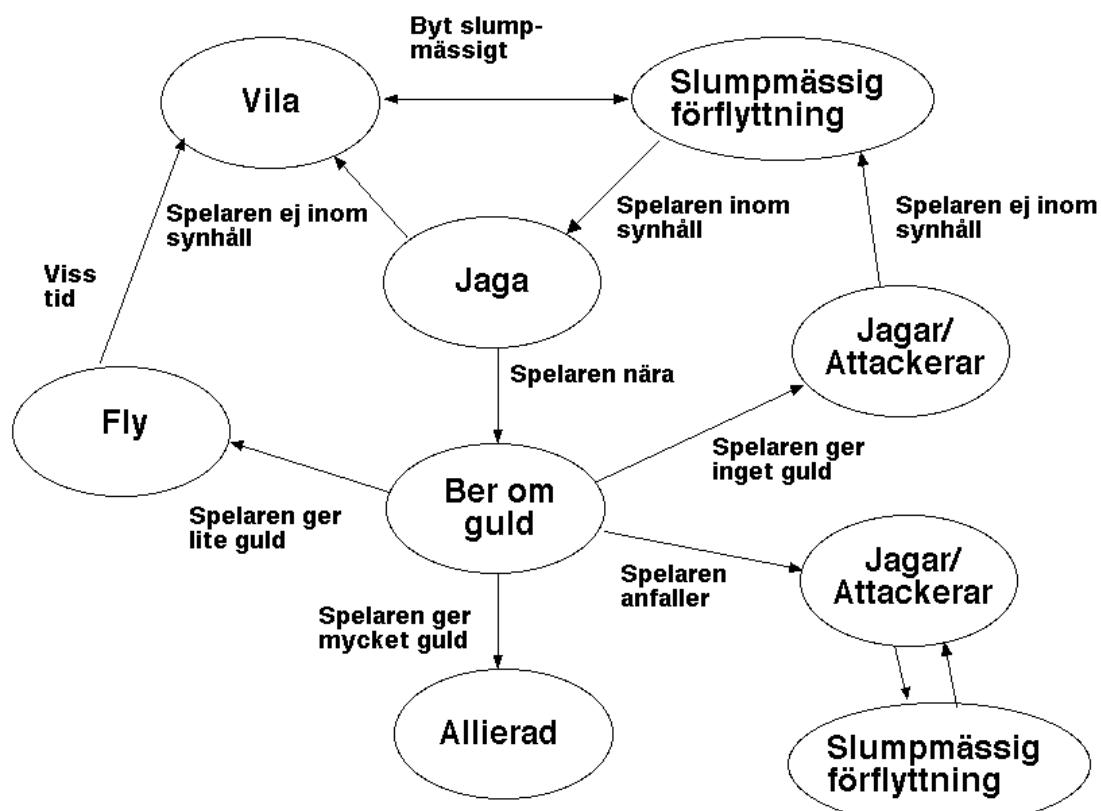
- Ljud i närheten
- Spelaren/intressant objekt inom synhåll
- Agenten skadas, angrips
- Kommunikation

Detta kan göras med en finita
tillståndsmaskin (FSM). Enkelt exempel:



Ingemar
Ragnemalm
ingis@isy.liu.se

Något mer avancerat exempel



Finita tillståndsmaskiner är ofta onödigt formella och onödigt stelbenta

Parallelta FSM ger mer frihet och är ofta enklare än FSM. Flera delbeteenden kan modelleras separat.

Regelsystem

Enkla regelsystem kan representera många action-AI

- Om jag inte ser någon spelare, vandra slumpmässigt
- Om jag ser en spelare, gå ditåt
- Om jag är nära spelaren, anfall

Ingemar
Ragnemalm
ingis@isy.liu.se

Fler action-AI-problem

Ögonkontakt

Hide&take cover

Jaga med prediktion

Skjuta (ev. med prediktion)

Ingemar
Ragnemalm
ingis@isy.liu.se

Världrepresentation

Viktigt problem för AI! Det viktigaste?

Många AI-problem lösas genom att välja världrepresentation

- Hur ser AI'n geometrin?
 - Hur lösas synlighet?
 - Ger vi för mycket information till agenterna, så de kan fuska?
 - Annan extra information?

Ingemar
Ragnemalm
ingis@isv.liu.se

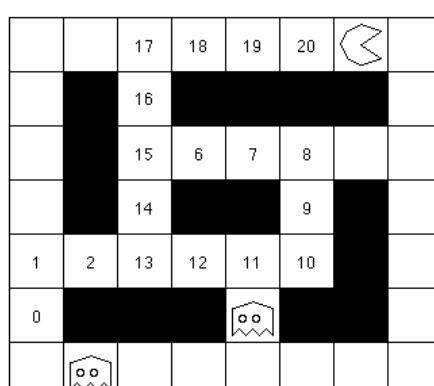
Världrepresentation

Exempel:

Pac-Man-liknande spel

Världen är en grid, representeras med array

Extra information: Spelaren lämnar ”doftspår” efter sig som fienderna kan följa



Ingemar
Ragnemalm
ingis@isy.liu.se

Världrepresentation

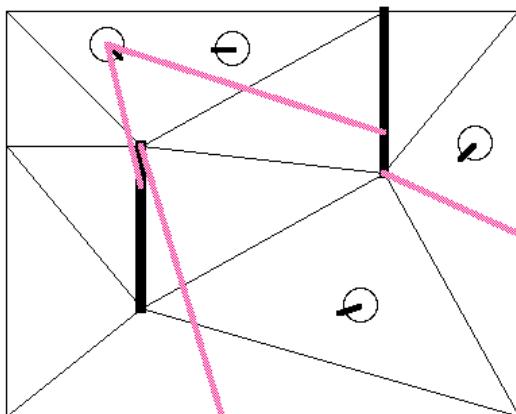
Exempel:

Robowar-likande spel men med hinder

Världen är en polygonmesh av trianglar

Trianglarna har pekare till alla grannar och pekare till alla objekt i triangeln

Synlighet i övriga trianglar kan bestämmas med portal-likande frustumberäkningar (men i 2D)



Ingemar
Ragnemalm
ingis@isy.liu.se

Avstånd

Världrepresentationen är ofta gridbaserad!

Diskret värld -> avstånd ofta icke-Euklidiska!

Exempel:

Schack

Sid Meier's Civilization

Ingemar
Ragnemalm
ingis@isy.liu.se

Metrik

En metrik är ett avståndsmått som uppfyller:

$$\begin{aligned}D(p_1, p_1) &= 0 \\D(p_1, p_2) &> 0, p_1 \neq p_2 \\D(p_1, p_2) &= D(p_2, p_1) \\D(p_1, p_3) &\leq D(p_1, p_2) + D(p_2, p_3)\end{aligned}$$

Ingemar
Ragnemalm
ingis@isy.liu.se

Metriker

Euklidisk: $D = \sqrt{(\Delta x^2 + \Delta y^2)}$

City Block: $| \Delta x | + | \Delta y |$

Chessboard: $\max(| \Delta x |, | \Delta y |)$

Viktade heltalsmetriker:

Chamfer 2-3
Chamfer 3-4
Chamfer 5-7-11

Viktade flyttalsmetriker

Chamfer 1- $\sqrt{2}$

Ingemar
Ragnemalm
ingis@isy.liu.se

Taktisk AI

Planering av förflyttning

“Crash & turn”

Dijkstras algoritm

A*

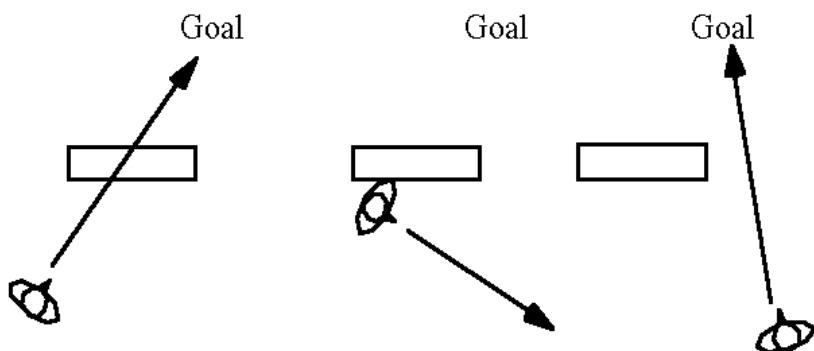
Omplanering när en väg blir oframkomlig

Är den optimala vägen rätt?

Ingemar
Ragnemalm
ingis@isy.liu.se

“Crash & turn”

- Gå mot målet tills nånting tar emot
 - Gå i slupmässig riktning
 - Upprepa

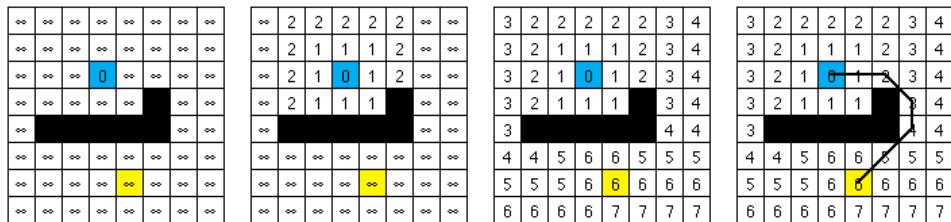


Fångit? Planlöst? Det beror på vilket beteende som agenten bör ha!

Ingemar
Ragnemalm
ingis@isy.liu.se

Kortaste-vägen-problemet

En geometri representerad som regelbunden grid: Löses med avståndstransform



(OBS! Vilken metrik används?)

A* lägger till heuristiska mått för att söka åt troligaste hålet först

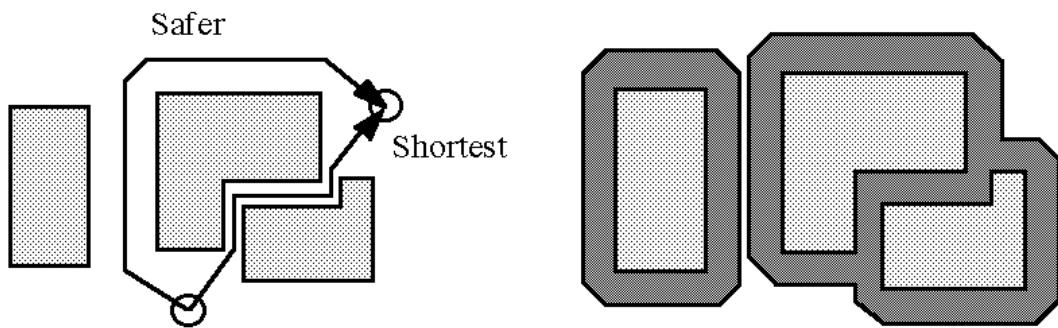
Men är detta egentligen AI? Och är det den kortaste vägen vi vill ha?

Ingemar
Ragnemalm
ingis@isy.liu.se

Säkraste vägen?

Prioritera vägar beroende på

- hur trångt det är
- faror, fiender...



Breda vägar kan hittas med matematisk morfologi

Ingemar
Ragnemalm
ingis@isy.liu.se

Sökning av tillståndsrymden

Minmax, Negamax

Metoder speciellt lämpade för deterministiska brädspel

Sök den där mitt bästa drag är bäst och motståndarens bästa drag är sämst

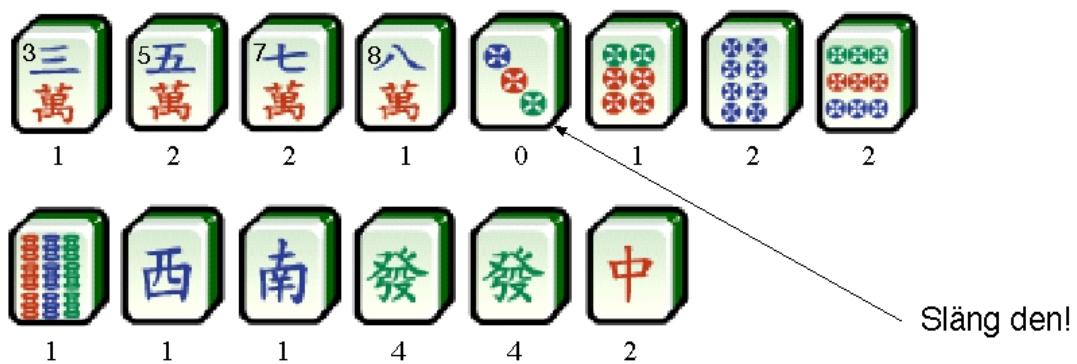
alpha-beta-pruning begränsar sökrymden till de delar som bedöms viktigast

Ingemar
Ragnemalm
ingis@isy.liu.se

Heuristisk utvärderingsfunktion

Lämpat för bl.a. icke-deterministiska brädspel

Gammalt favoritexempel:



- +1 för honör, drake, vind
- +2 om drake eller egen vind
- +2 för varje likadan (möjlig pong)
- +1 för granne inom två steg (möjlig chow)

Influence maps

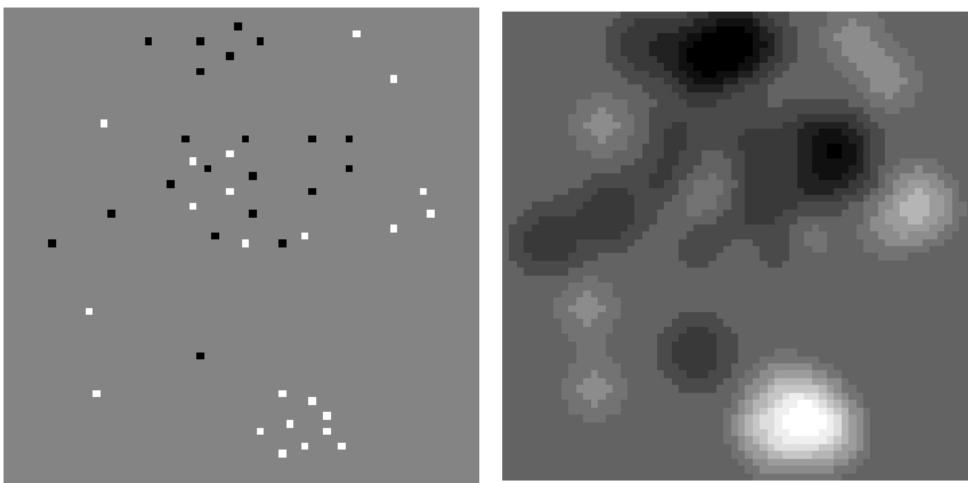
Kraftfull representation av information om spelsituation!

Lågpassfilter av karta av spelagenter förenklar analysen.

- Vem “äger” rutan?
- Är området säkert, inga fiender nära?
- Vilken annan medspelare står fri?
- Var är motståndaren som svagast?

Ingemar
Ragnemalm
Inglis@isy.liu.se

Influence maps, exempel



Varianter

- Terrängberoende filtrering, längsammare spridning i svår terräng
 - Flera influence maps för olika data
 - Olika värde i kartan för olika pjästyper

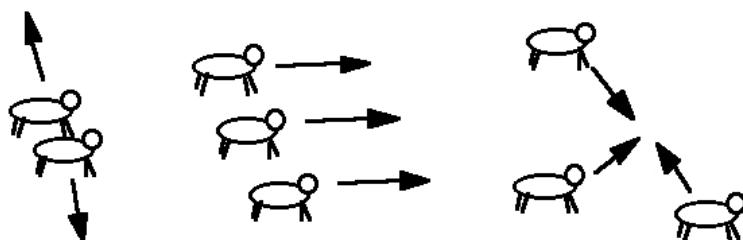
Varianterna styrs i stort av speltypen, men grundprincipen är generellt användbar till många problem.

Flocking - Boids

Grupp beteenden kan simuleras med ganska enkla medel.

Craig W Reynolds "boids", baserade på tre enkla regler:

- Separation - undvik att gå för nära varandra
 - Alignment - sträva åt samma håll
- Cohesion - sträva mot mitten av gruppen



Lärande system - ämnet jag fuskade bort i boken

System som lär sig är ett intressant ämne! Mest kända metoderna:

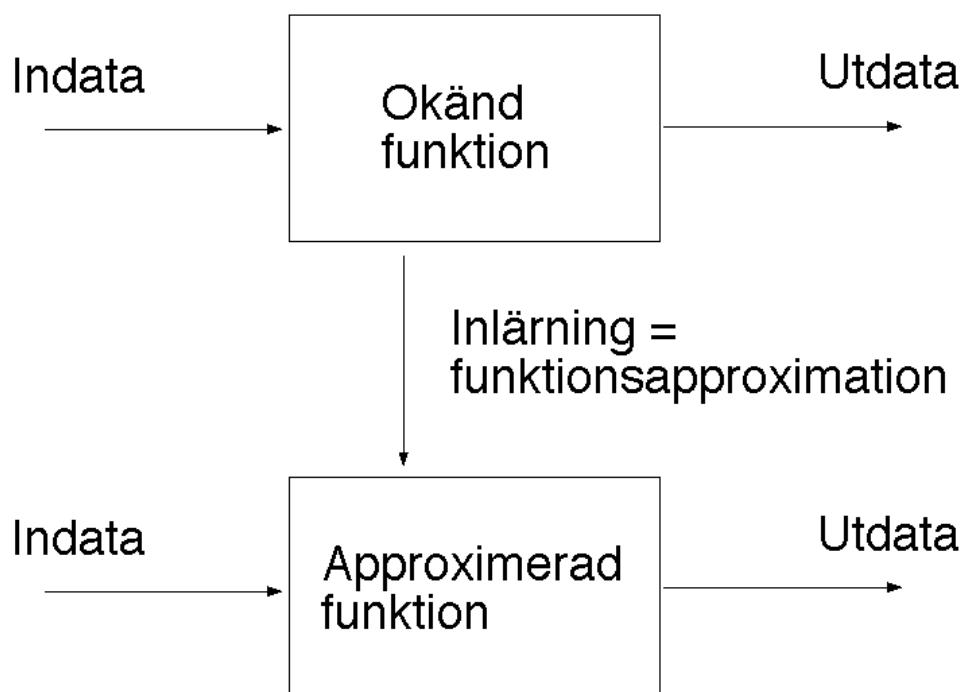
- Nervnät
- Genetiska algoritmer

Aningen mindre kända:

- Simulated annealing
 - Hill climbing
- Heuristiska metoder

Men observera: Allt är i grund och botten *optimering!*

Optimering?



Funktionsapproximation är inte alls tråkigt, men...

Mitt dilemma:

Är detta relevant för spel?

Både jag och Millington säger *nej*. Inlärning har en ytterst liten roll i spel!

Heuristiska metoder troligen intressantast, för begränsad inlärning hos agenter.

Nervnät mm har framgångsrikt använts på bl.a. bildanalys, men knappast på spel.

Spel med inlärning eller liknande inslag:

SimLife

Viss simulering av genetik, men

- urval av population är inte (exakt) samma sak som inlärning och upplevs inte som det
- Spelet var inte särskilt underhållande och simuleringen var dålig

Black&White

Inlärning av beteende på "varelsen"

Intressant idé, mindre lyckat resultat

Men är AI viktigt för spel?

Javisst!

- Beteenden och vägplanering
 - Världsrepresentation
- Analys av speldata i denna representation