

TDDD92 Artificiell intelligens - projekt (2024)

Problemområden

Daniel de Leng

Postdoktor

Institutionen för datavetenskap (IDA)

Hur gick det?



Introduktion och motivering

Ni ska inför er individuella uppgift **definiera ett problem** och **välja ut en eller flera AI-tekniker** som kan tänkas användas för att lösa detta problem.

- Det finns väldigt många domänspecifika **problem** att lösa. Man behöver inte lösa alla, utan välja sådana som kan bidra till en bra agent.
- Det finns väldigt många **AI-tekniker**. De kan ersätta varandra, komplettera varandra eller integreras för att lösa större problem.

Vi använder exempelscenarier från en gammal *StarCraft II* match

URL: <https://www.youtube.com/watch?v=LtF1-x1ipU&t=3320s>

Vad vi inte gör idag

Idag ska vi **inte**:

- Diskutera och definiera alla **problem i detalj**:
=> Ni ska själva utforska och skapa egna varianter av dessa generella problem.
- Diskutera och specificera **lösningstekniker i detalj**:
=> Vi ger tips, ni utforskar lösningar.

Problem och tekniker

Byggnadsplacering

När du bygger nya byggnader, och **var** ska de placeras?

- Kan innebära geometriska resonemang
- Måste tänka på hur kartan ser ut – anpassa sig till olika kartor

Möjliga mål:

- Undvika att blockera de egna enheterna – låt dem röra sig fritt i er bas.
- Blockera platser där fienden kan tänkas vilja röra sig.
- Bygga murar för att göra det svårare för fienden att ta sig in i er bas.
- Hitta platser för flera baser för decentraliserad produktion.

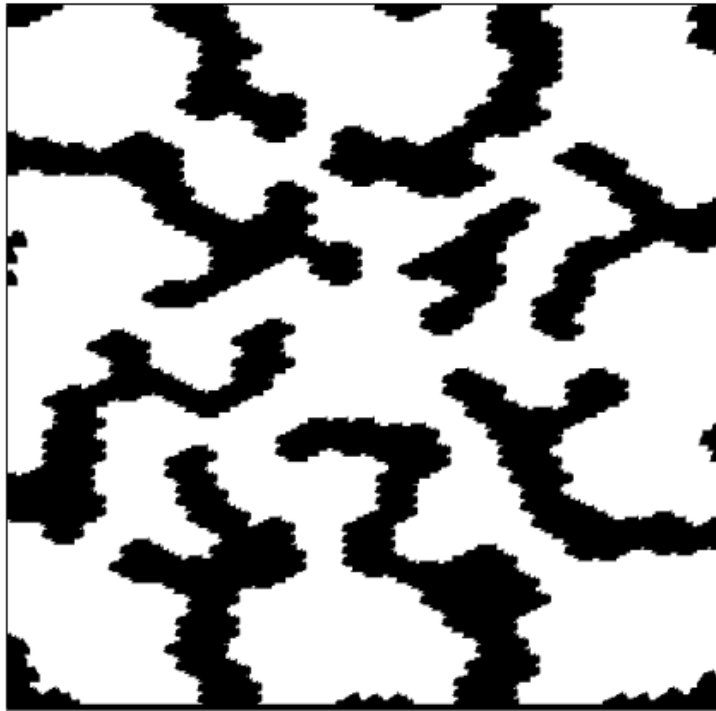
Byggnadsplacering

Vi kan tänkas vilja veta var det finns:

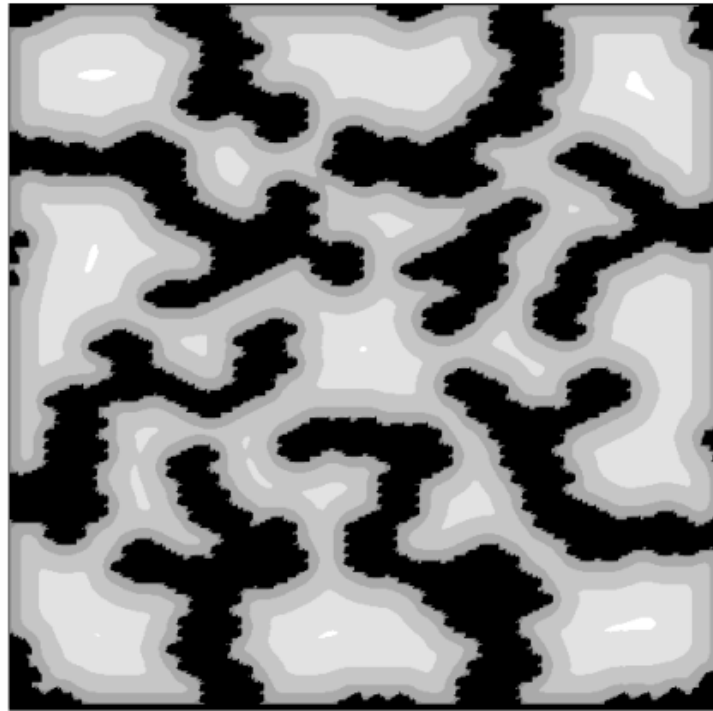
- **Fria regioner** för t.ex. specifika byggnader eller kompletta baser.
- **Trånga områden** för t.ex. försvar.

Decomposition

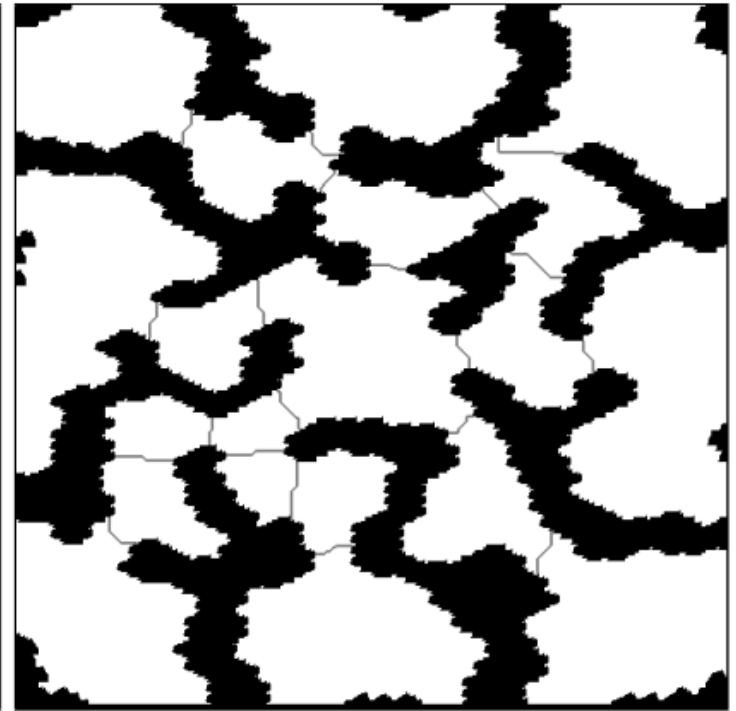
Hitta flaskhalsar (*choke points*)



Gångbar terräng

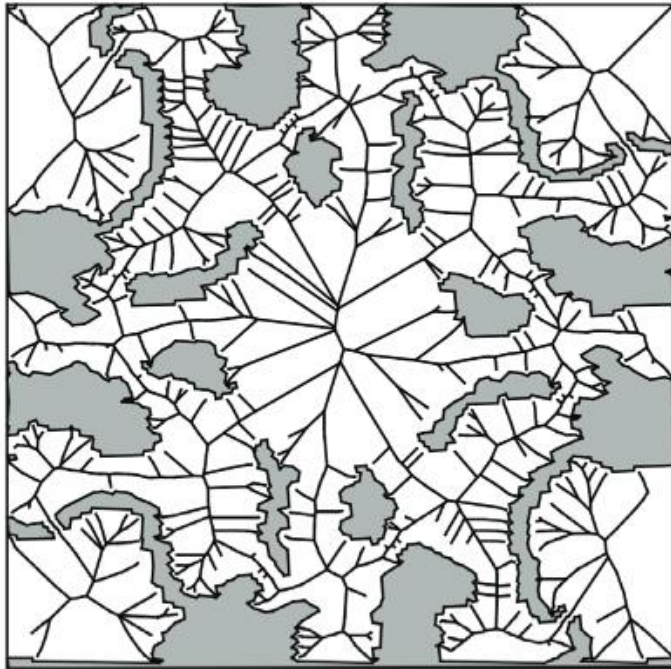


Flood fill

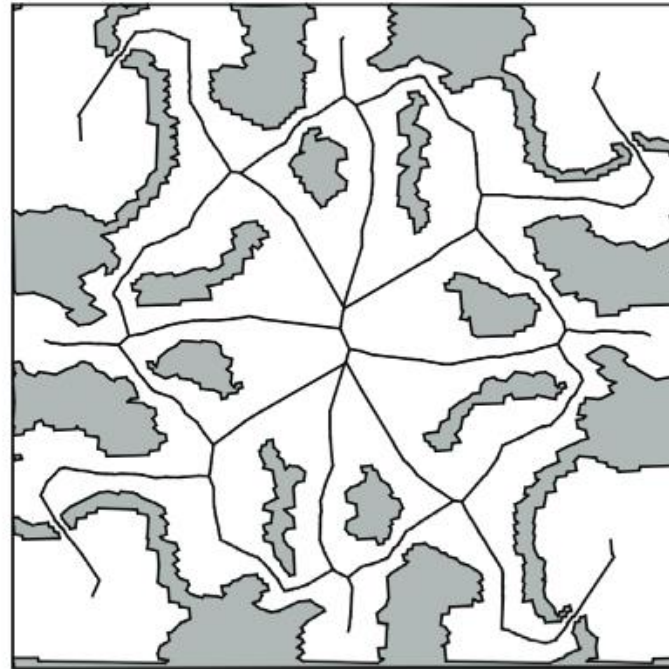


Flaskhalsnoder

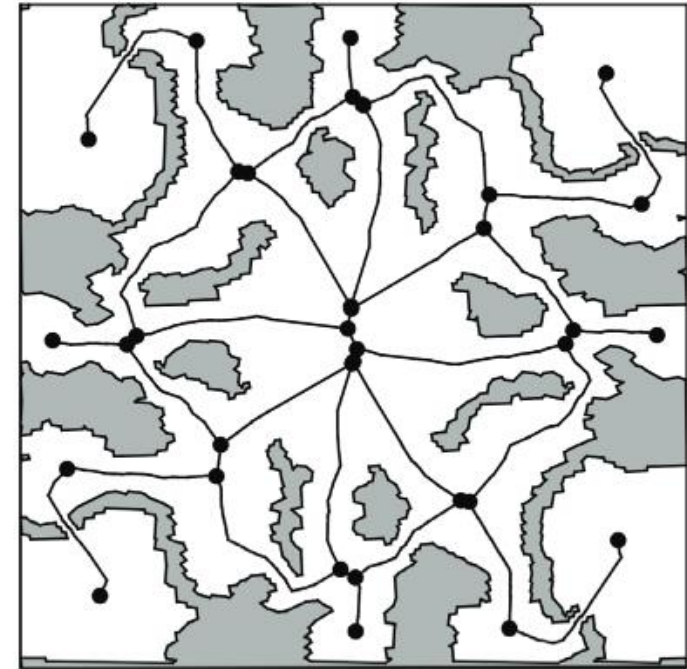
Hitta flaskhalsar (*choke points*)



Voronoi-diagram



Efter beskärning (*pruning*)



Upptäckta regionnoder



Production [D] ✓

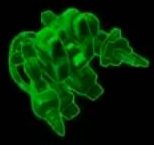
17/27 17/18

218 260 17/27 17/18

3:03

3:03 Faster
Everyone (E) ? 3 Menu





45 / 45

SCV

Kills: 0
Rank: Recruit



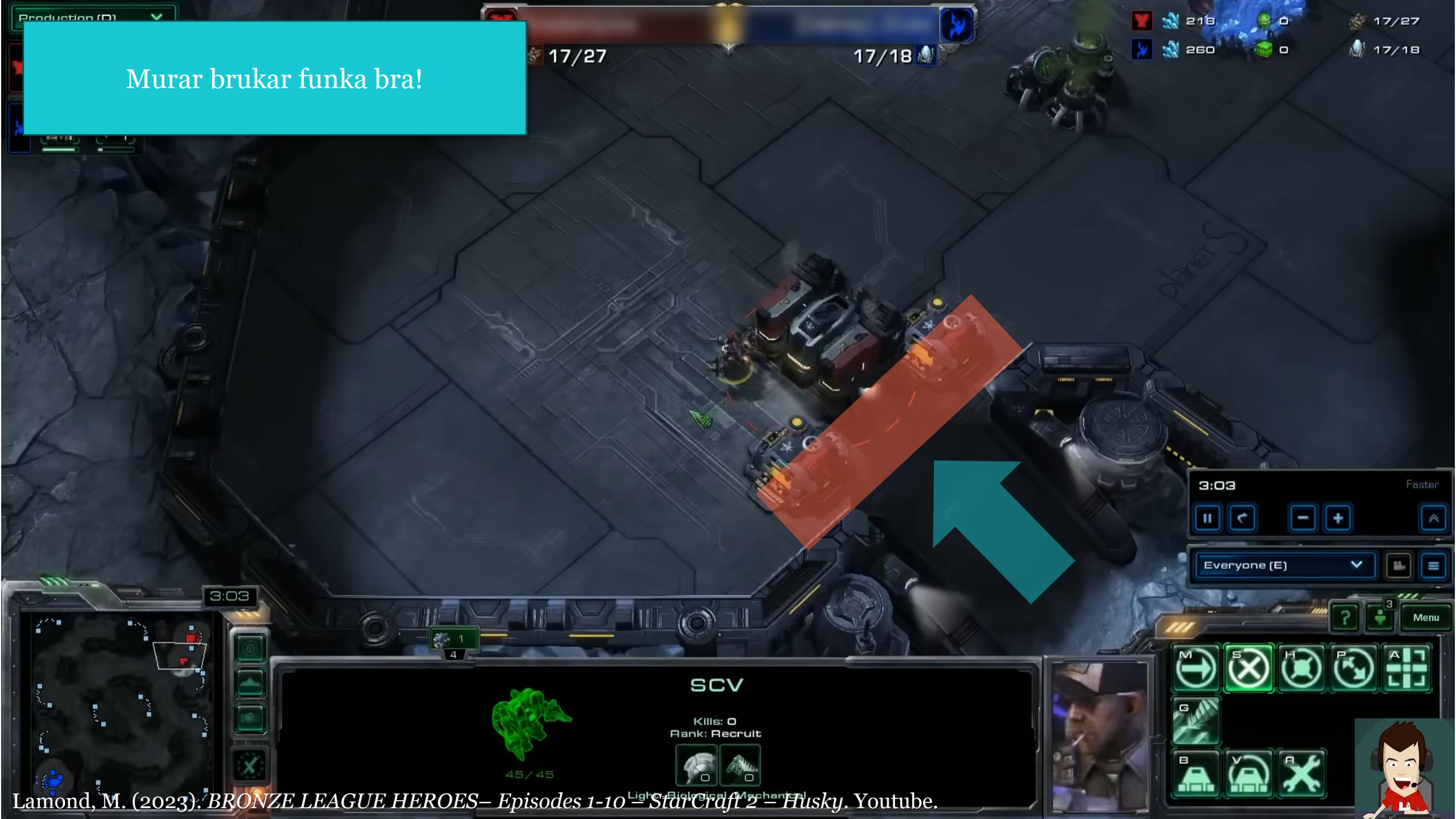








Murar brukar funka bra!





Production (D) ✓

6

3

200/200

189/200

3960

967

200/200

4224

2241

189/200

Workers: 0/18

23:51 / 25:34

Faster

|| ◀ - + ▶ ▾

Everyone (E) ▾

?

3

Menu

23:51

?

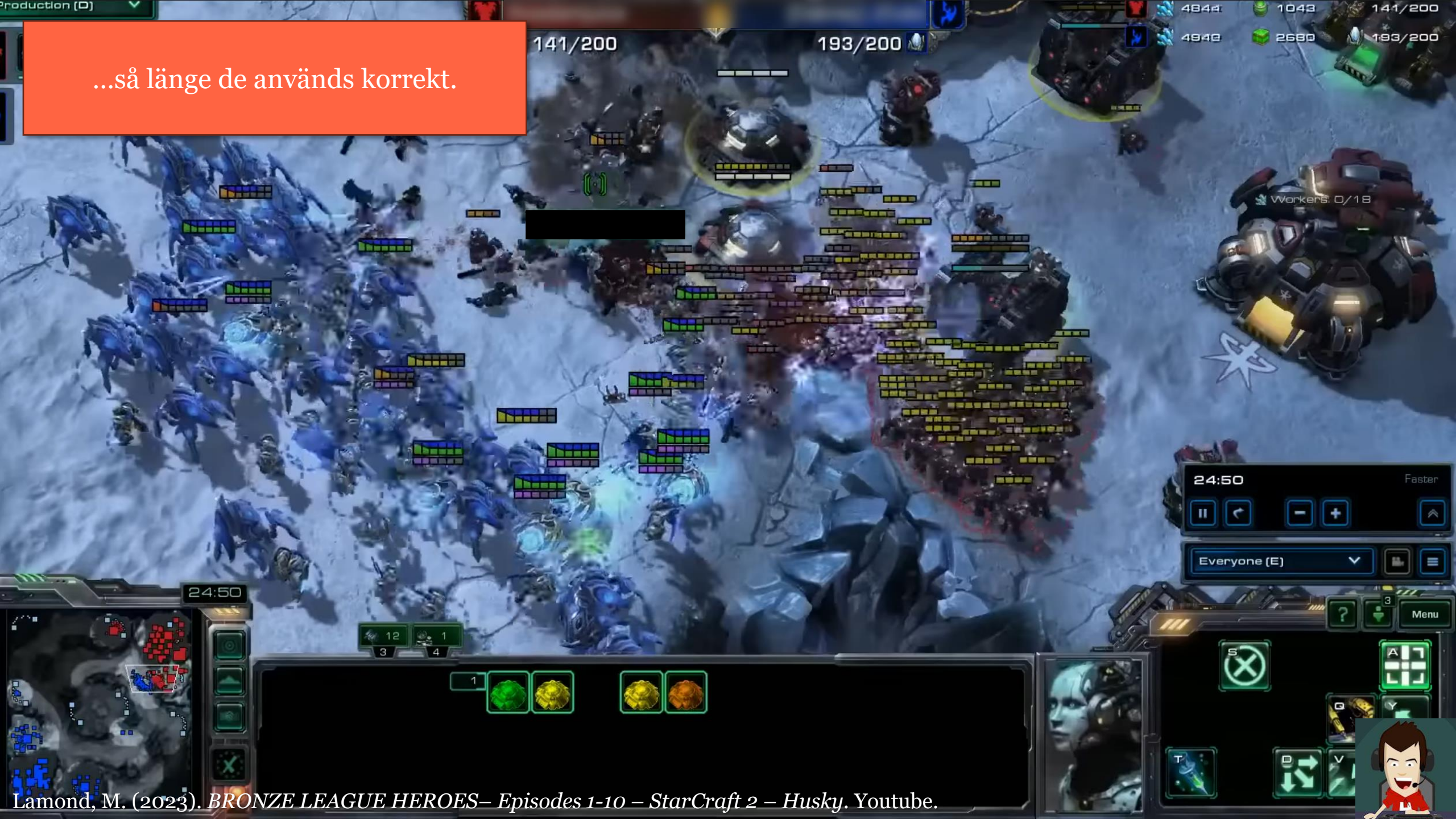
3

Menu

Flaskhalsar kan också
förbättra försvar...

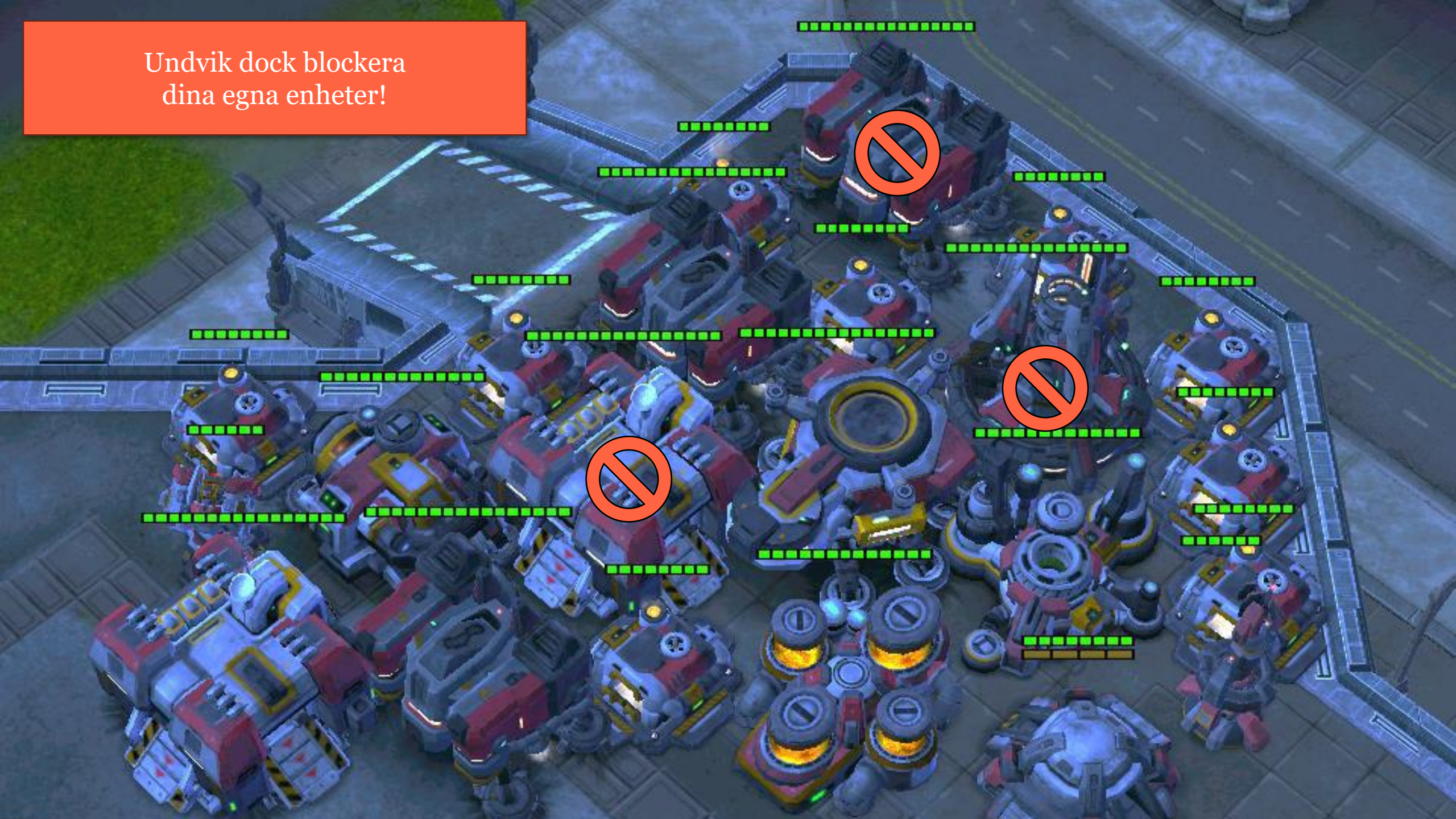


...så länge de används korrekt.





Undvik dock blockera
dina egna enheter!



Produktionsplanering

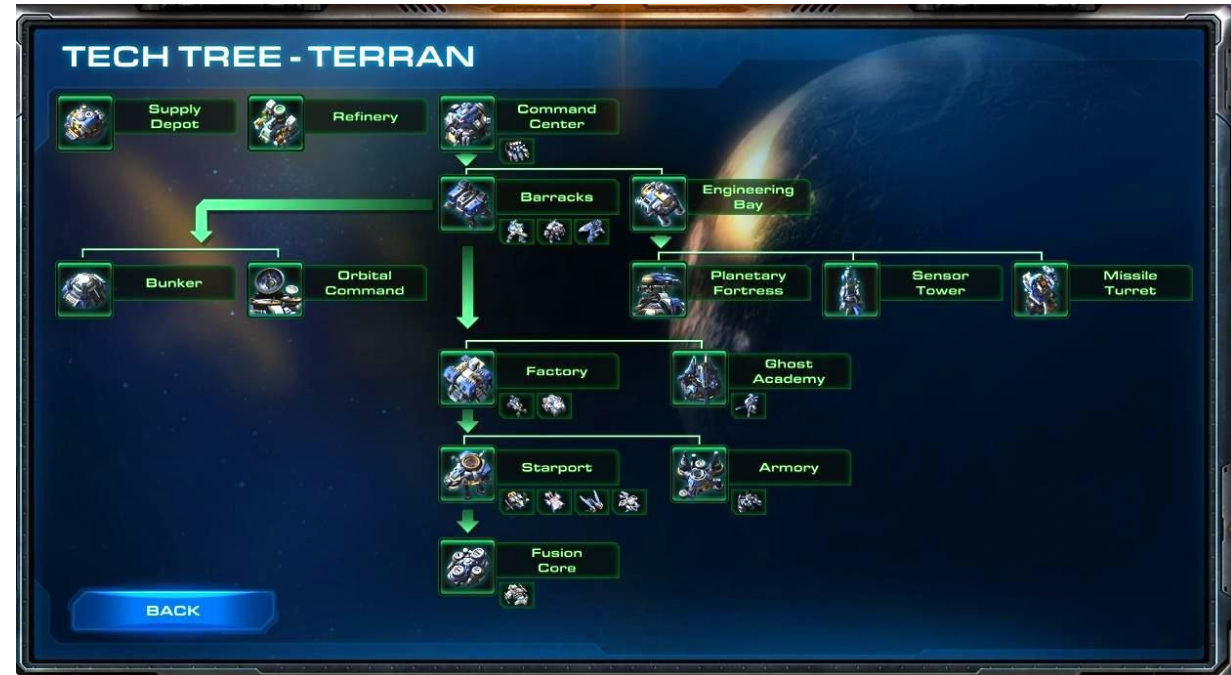
Vad ska byggas / produceras / uppgraderas, och i vilken **ordning**?

Tänkbara konkretiseringar:

- **Hur uppnår vi ett visst produktionsmål på bästa sätt?**
Input: Produktionsmål, budget/inkomst, nuvarande byggnader, etc.
Output: Bästa möjliga plan/policy för att uppnå produktionsmålet.
- **Vilken produktionsplan är bäst för att kontra fiendens strategi?**
Input och lösning (algoritm) behöver också ta hänsyn till motståndaren, t.ex. bygga försvar mot anfall eller för att kontra flygande enheter.

Produktionsplanering

- Teknologiträden visar i någon mening vad man behöver först (*dependencies*): $A \prec B$
- Att **bara** titta på teknologiträden är för trivialt – även om man t.ex. använder en sökalgoritm.



Produktionsplanering

Måste ta hänsyn till mer!

Exempel:

- Hur många enheter vill vi bygga?
- Vilka resurser har vi tillgängliga?
Vilka kommer vi att ha tillgängliga när vi bygger nästa steg?
- Hur lång **tid** tar det att genomföra alternativa produktionsplaner?



Produktionsplanering

Teknikexempel: Sökning, planering

- Bygg upp en lämplig **modell** av världens **tillstånd** och av de **handlingar** som kan utföras.
- Sök genom en tillståndsrymd: Vad kan man göra? Vad händer då?
- Hitta en plan som leder till att man har det som önskades.

Time	Action	Supply
0:17	 Supply Depot	14 / 15
0:24	 SCV	15 / 15
0:38	 SCV	16 / 23
0:40	 Barracks	16 / 23
0:44	 Refinery	16 / 23
0:51	 SCV	17 / 23
0:53	 Refinery	17 / 23
1:03	 SCV	18 / 23
1:15	 SCV	19 / 23
1:27	 Reaper	20 / 23

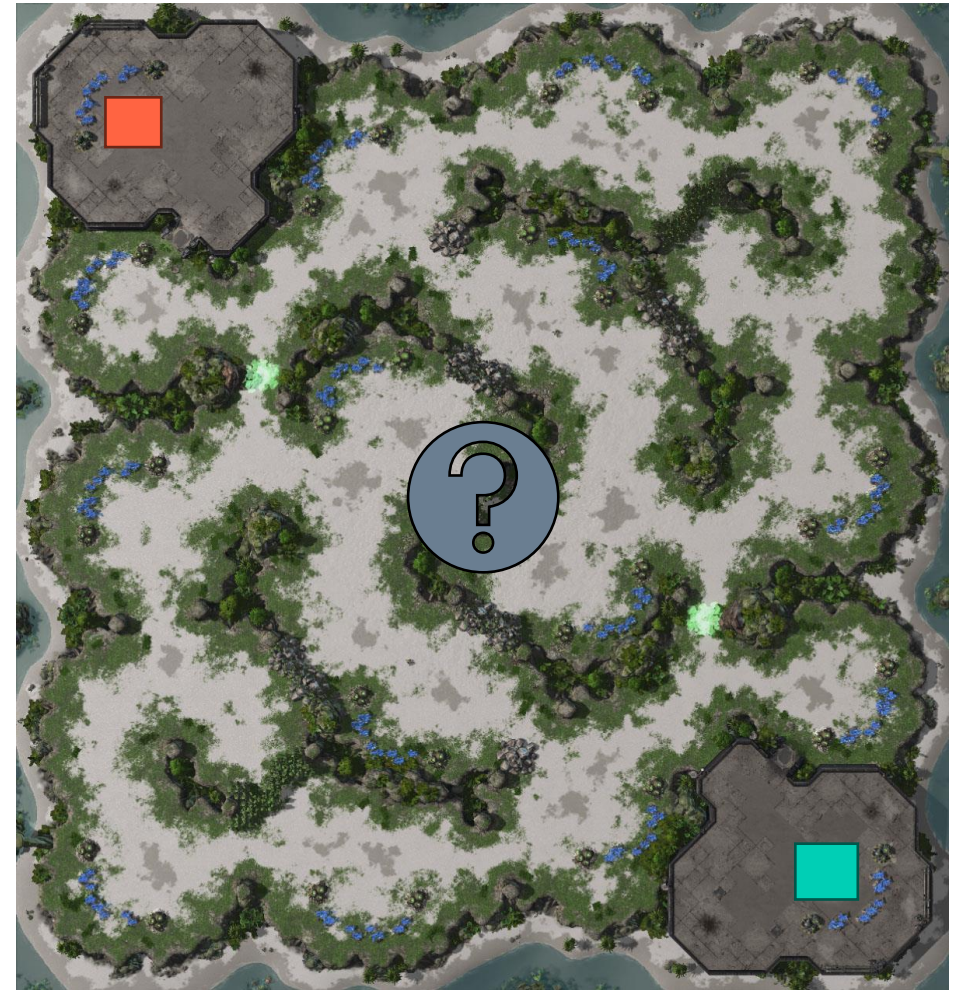
Utforskning av kartan

Givet **prioriteringar**, **mål** och **nuvarande kunskap**:

- Vilka delar av kartan bör vi utforska?
- När bör vi utforska?
- Och i vilken ordning?

Input: Tidigare observationer av motståndarens enheter; kartan.

Output: En sekvens av positioner på kartan som scouten skall röra sig till.

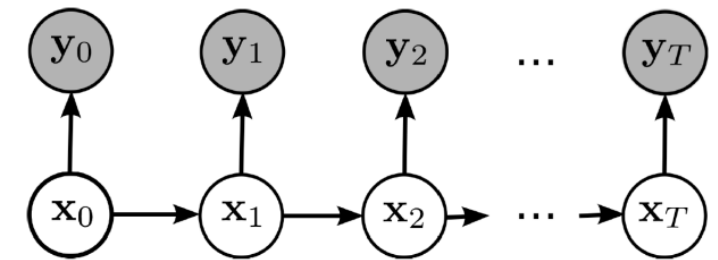


Utforskning av kartan

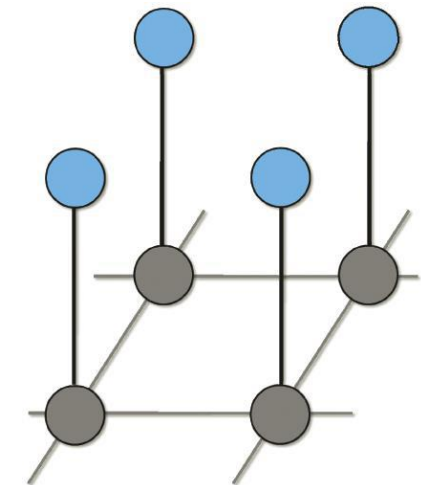
Hidden Markov Models (HMMs): Givet vad **ni observerar just nu** och vad **ni har observerat tidigare**:

$$\begin{aligned}\Pr(\text{motståndarBas} \mid \text{basPlats}) &= ? \\ \Pr(\text{motståndarenheter} \mid \text{position}) &= ? \\ \Pr(\text{motståndarstridkraft} \mid \text{position}) &= ?\end{aligned}$$

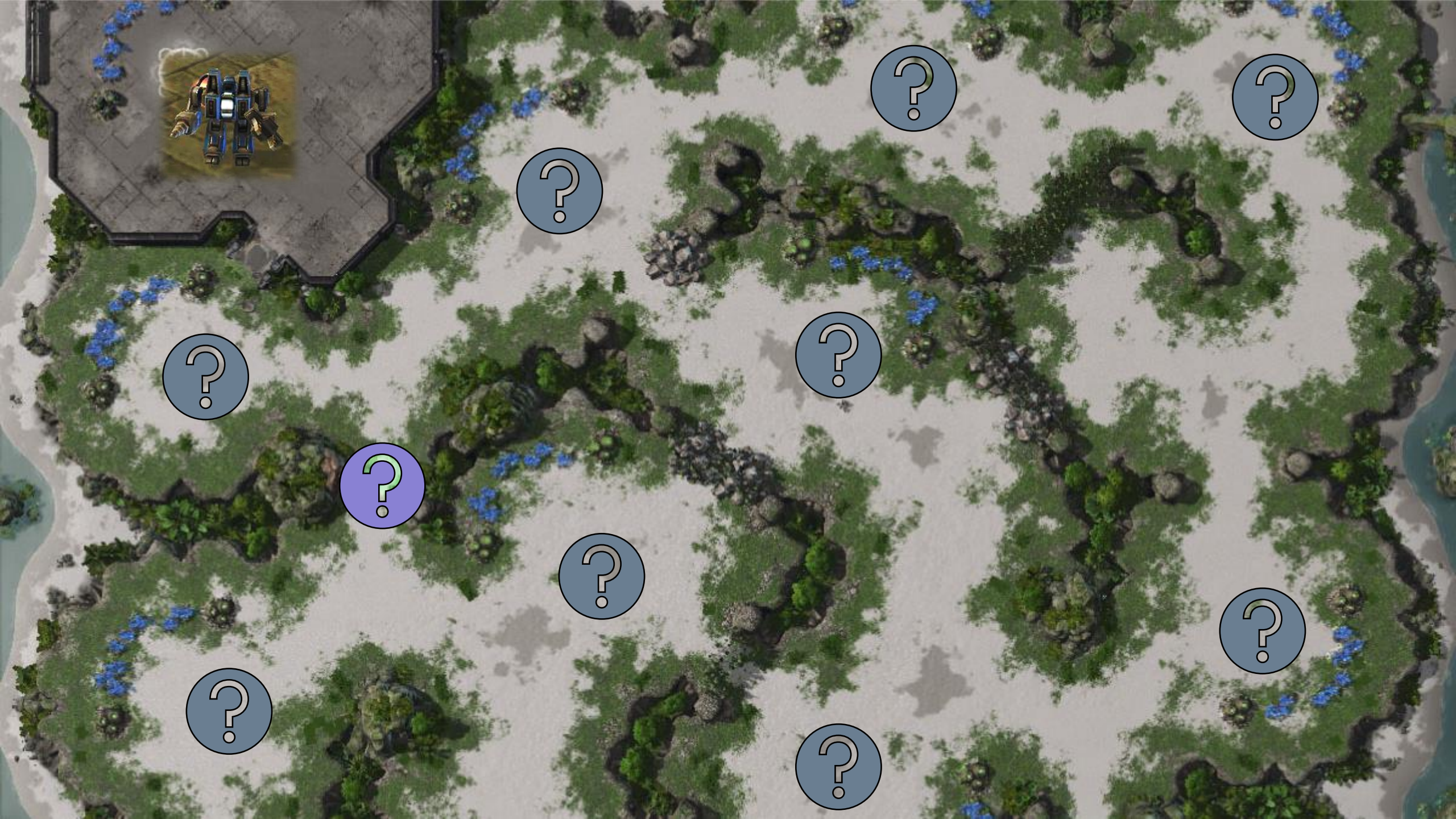
"När utforskades plats $\mathbf{x}$ sist?" vs "Hur troligt är det att det finns något nytt att observera på plats $\mathbf{x}$?"



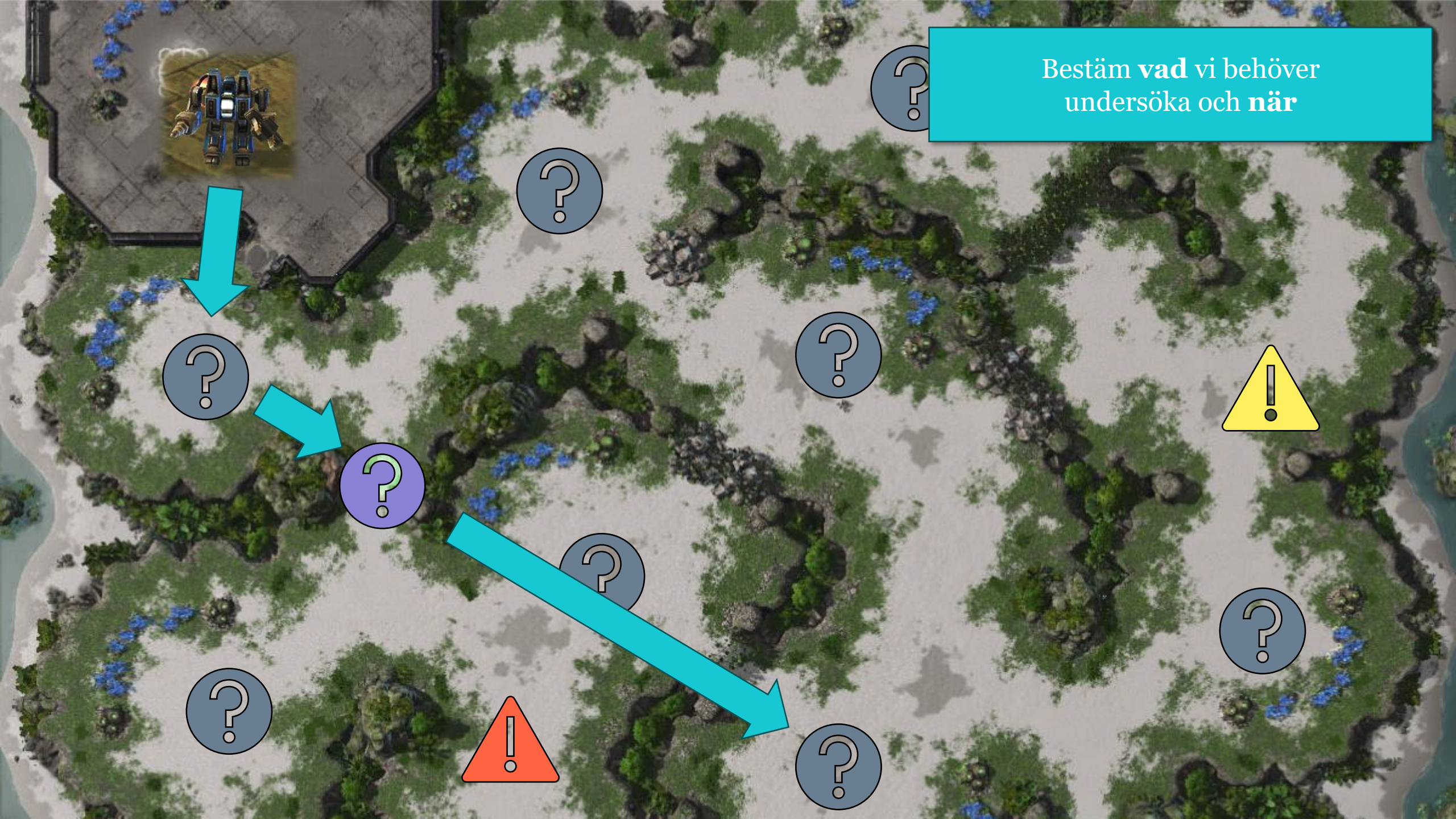
Exempel på temporal HMM.



Exempel på spatial HMM (på en grid).



Bestäm **vad** vi behöver
undersöka och **när**





Production (0)

196/200 158/158

1372 233 196/200
810 633 158/158

Unit selection panel showing icons and counts for various units.

Minimap showing the game area with a red line indicating a path.

Unit information panel for the Immortal unit, showing stats and abilities.

Command panel with a chat window and a player avatar.

Lamond, M. (2023). BRONZE LEAGUE HEROES- Episodes 1-10 - StarCraft 2 - Husky. Youtube.

Blå upptäcker en röd marinsoldatarmé



Blå upptäcker en röd marinsoldatarmé



Truppförflyttning

StarCraft har en inbyggd funktionalitet för att gå från A till B

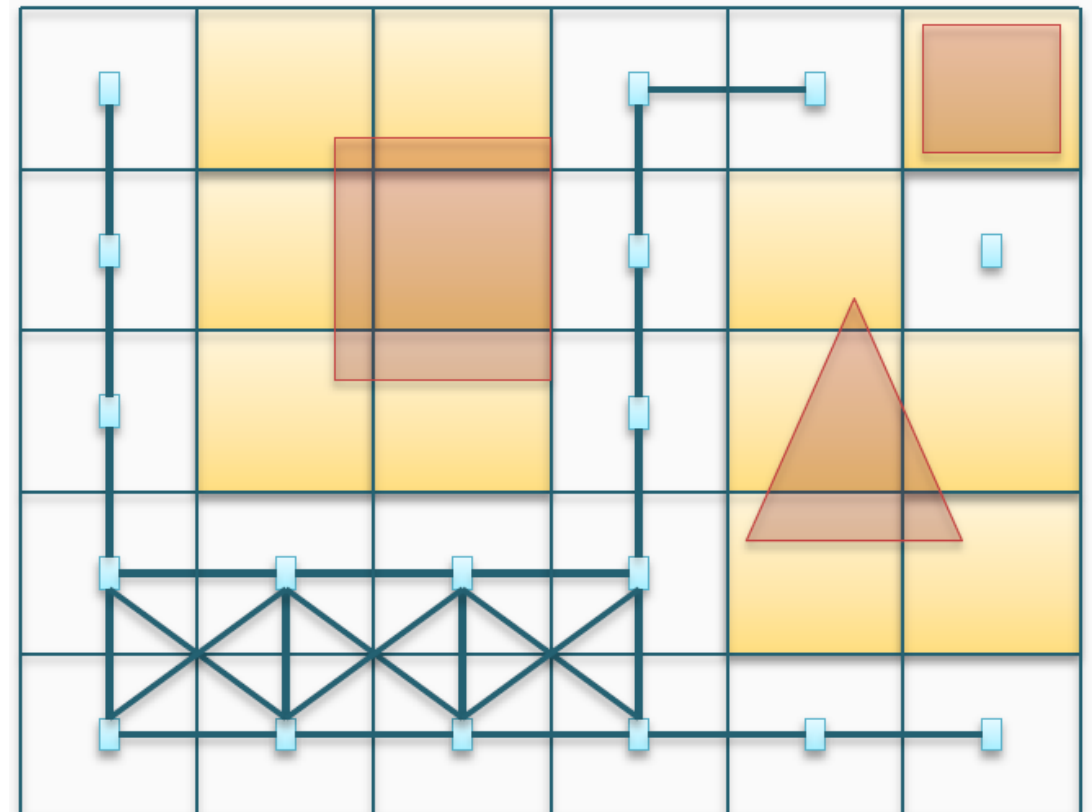
- Den tar **inte** hänsyn till var fiender finns.
- Undvik beskjutning? Håll dig utanför fiendens räckvidd.
- Undvik att bli upptäckt? Håll dig ännu längre bort.

Kan kräva **underrättelseinformation** från spaning...

Hitta säker väg

Exempel: Sökning

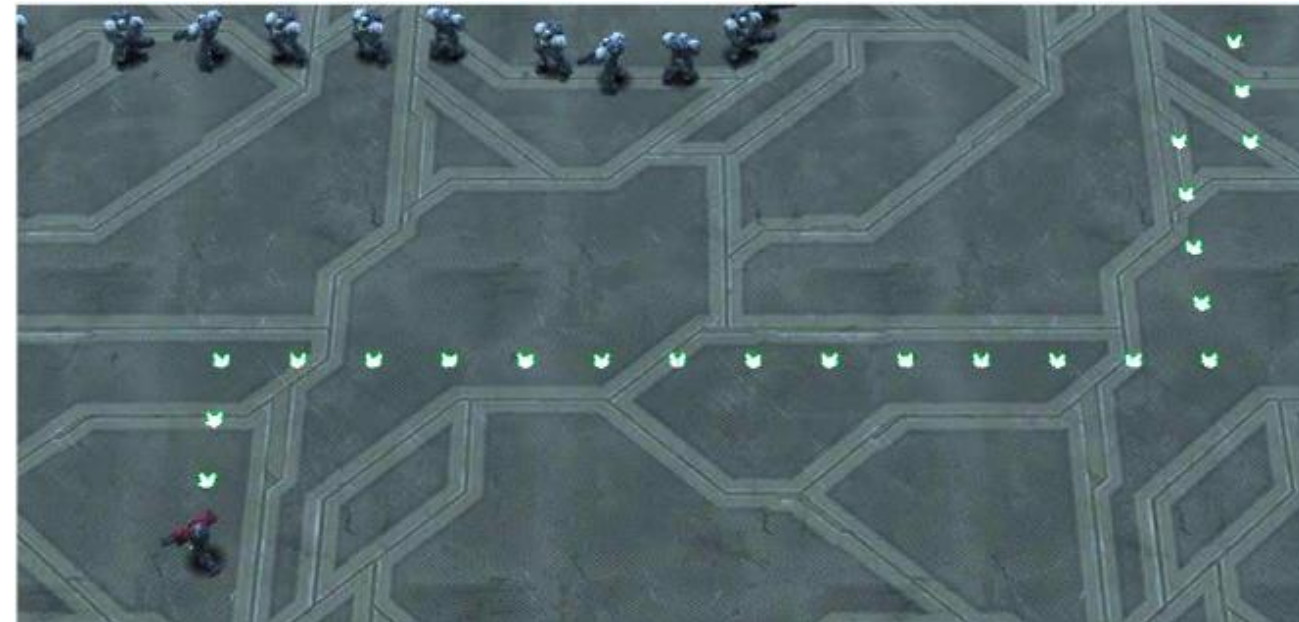
- Kan skapa **noder** motsvarande ett **rutnät**.
- Koppla ihop närliggande noder med **bågar**.
- Sök en väg genom rutnätet (med t.ex. A* och en rimlig heuristik).

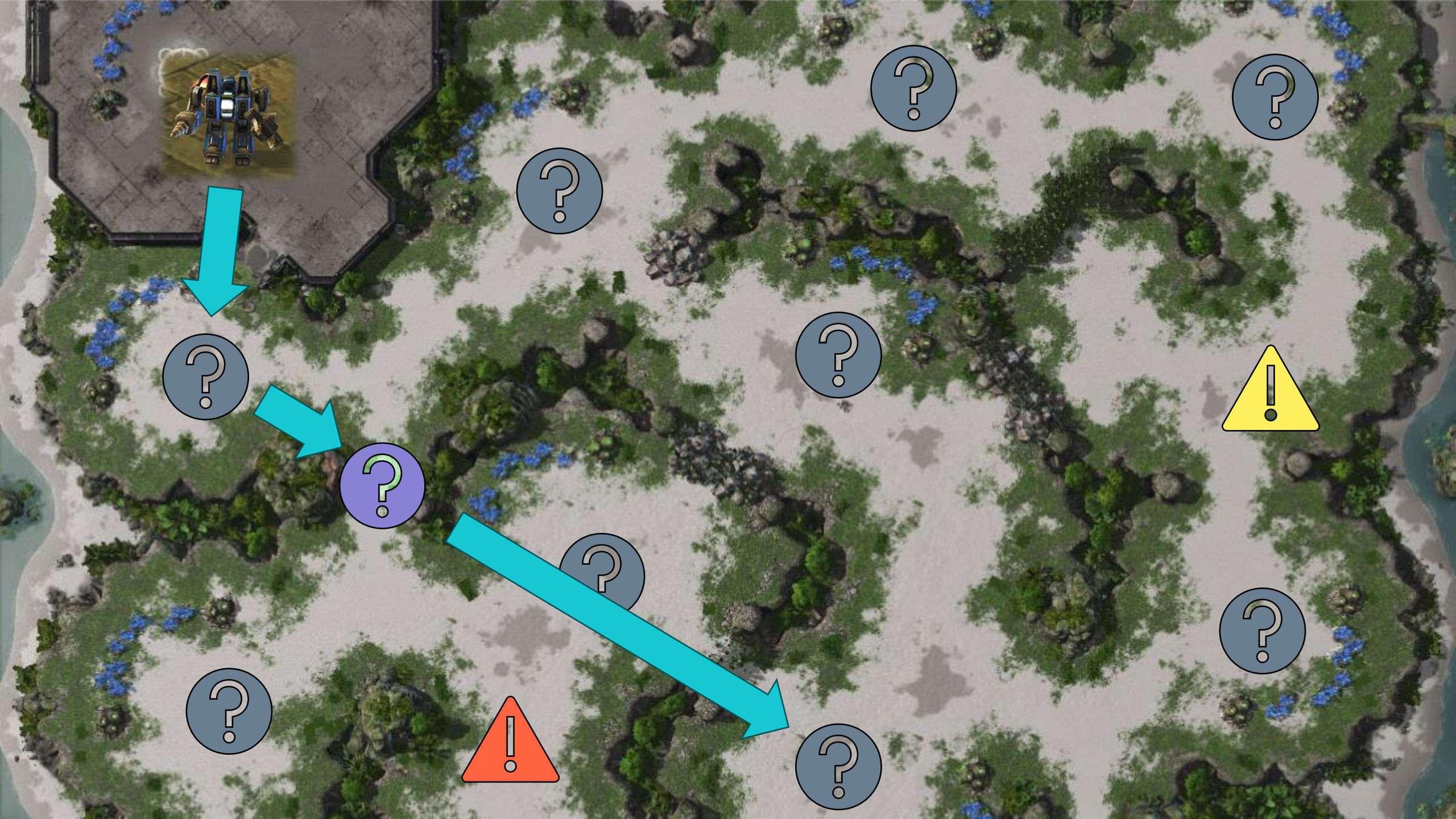


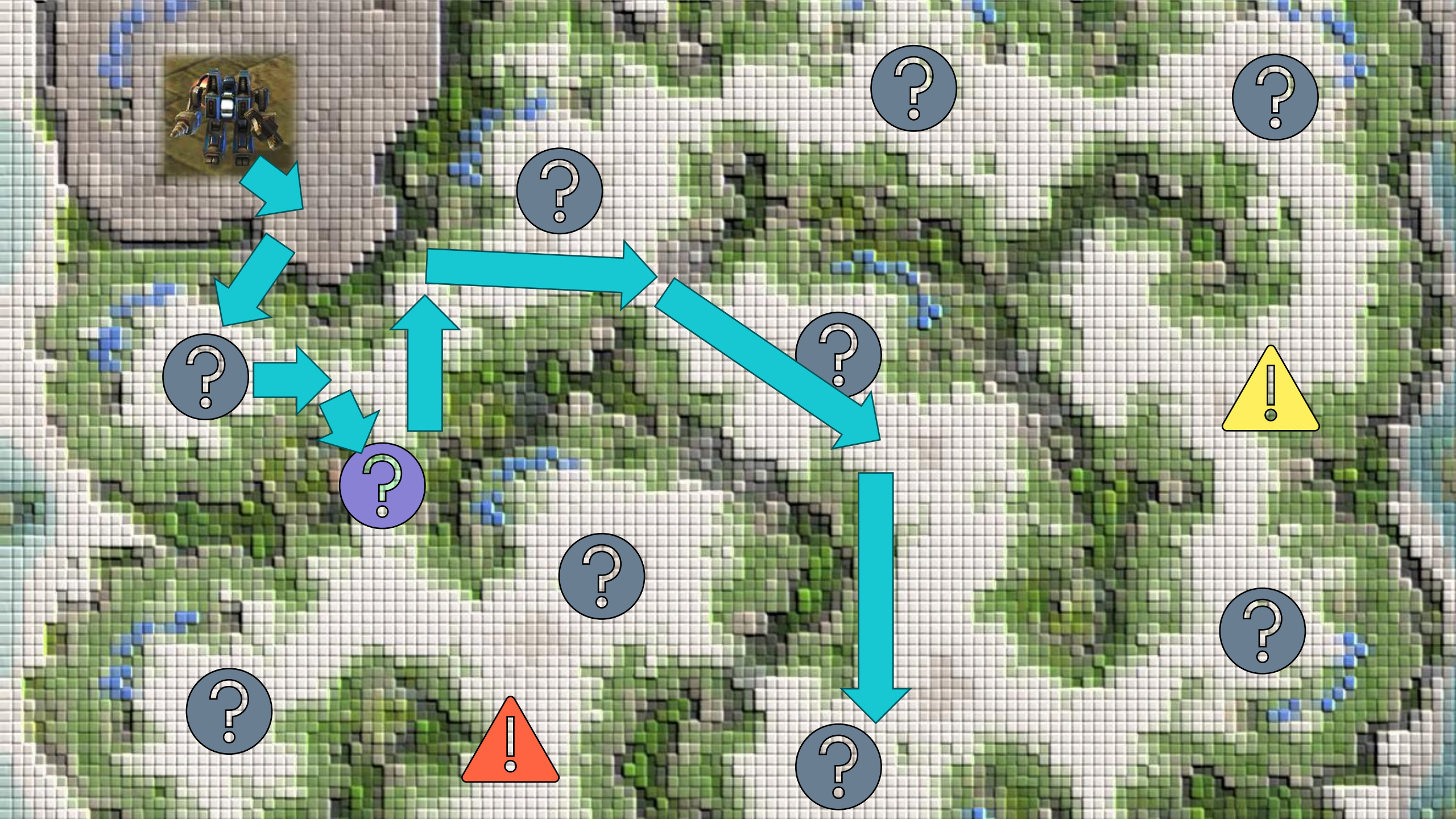
Hitta säker väg

Teknikexempel: A* / Dijkstra-sökning, potentialfält, flödesfält

- **A*** med bågkostnader proportionell mot risken att bli skadad i ett steg.
- **Potentialfält** som ger en kostnad / belöning att vara i en viss punkt.
- **Flödesfält** som attraherar till ett mål och repellerar från fienden.



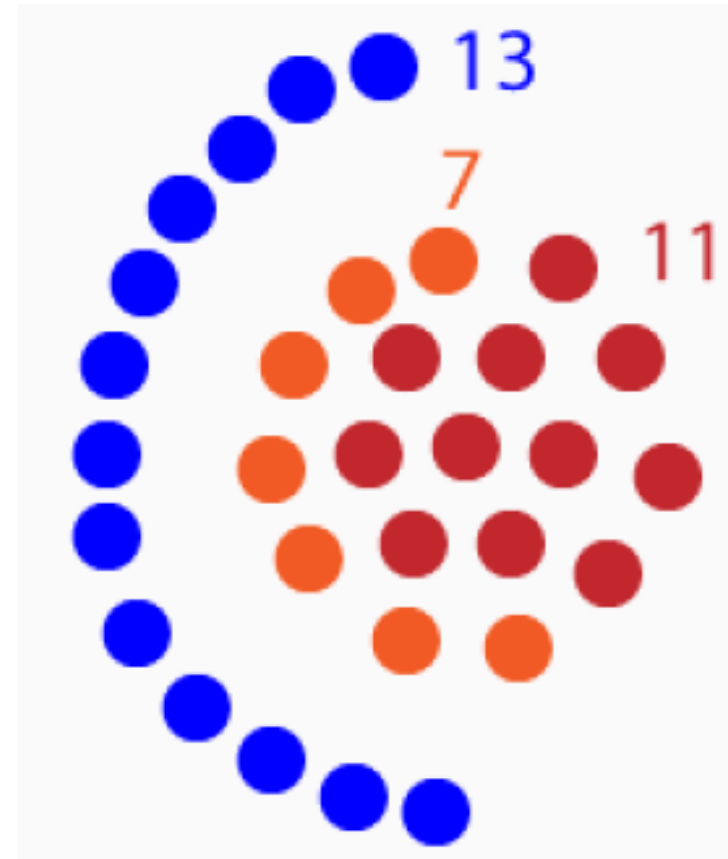




Truppfformationer och attackmönster

I många lägen behöver man detaljstyra positionen för medlemmar av en grupp:

- Exempel: **Truppfformationer**, t.ex. att attackera i en konkav båge.
- Exempel: **Mikromanagement** fokuserad eldgivning, *kiting*, utnyttja styrkor/svagheter, ...



Trupppositioner och attackmönster

Teknikexempel: *Flocking/Swarming* genom flödesfält/potentialfält

Olika beteenden är bra för olika taktiker:

- Utdela maximal skada/tid enhet
- Undvika skada (*range, melee, AoE*)
- *Marching orders*

Attraktorer och repellerare

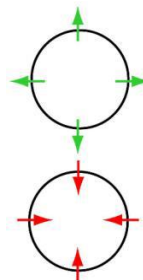
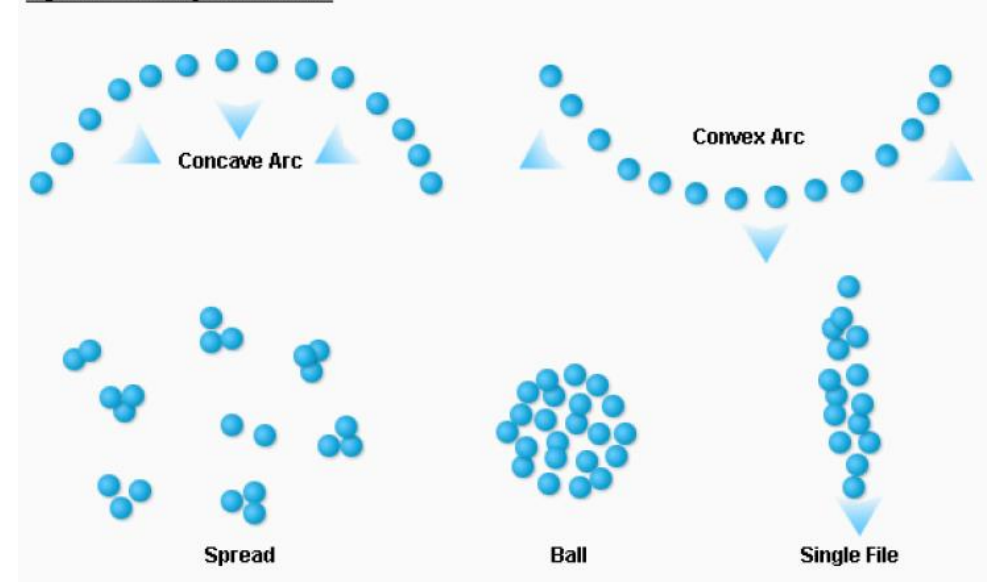


Figure A. "Primary Formations"



Källa: <http://www.teamliquid.net/forum/sc2-strategy/187892-positioning-formations-and-tactics>



Production (D) ✓

200/200 160/198

1834 276 200/200
1381 868 160/198

1

1

3

1

1

17:01 Faster

Everyone (E)

17:01

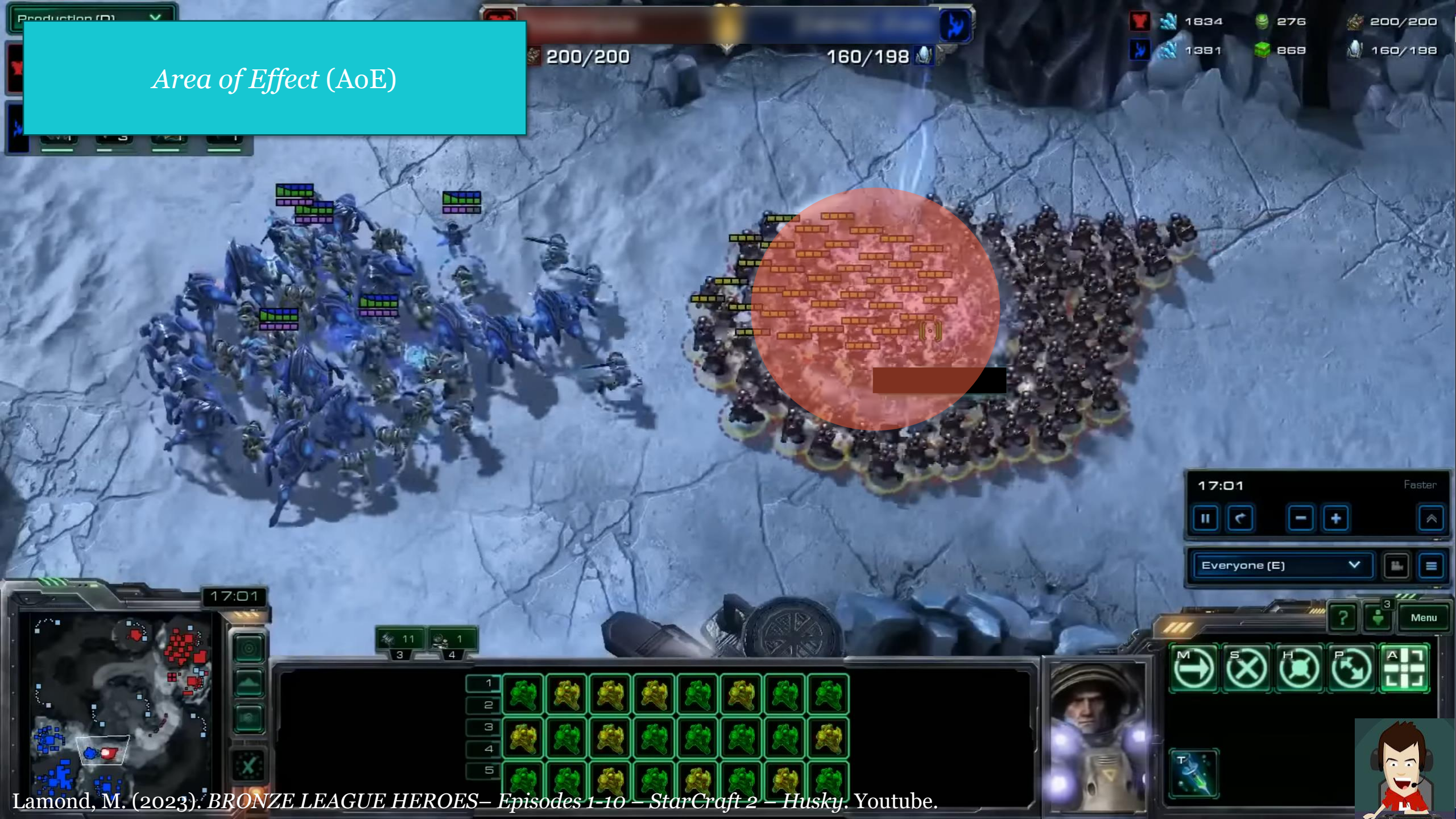
11 1
3 4

1								
2								
3								
4								
5								

3 Menu

Lamond, M. (2023). *BRONZE LEAGUE HEROES – Episodes 1-10 – StarCraft 2 – Husky*. Youtube.

Area of Effect (AoE)



Positioning

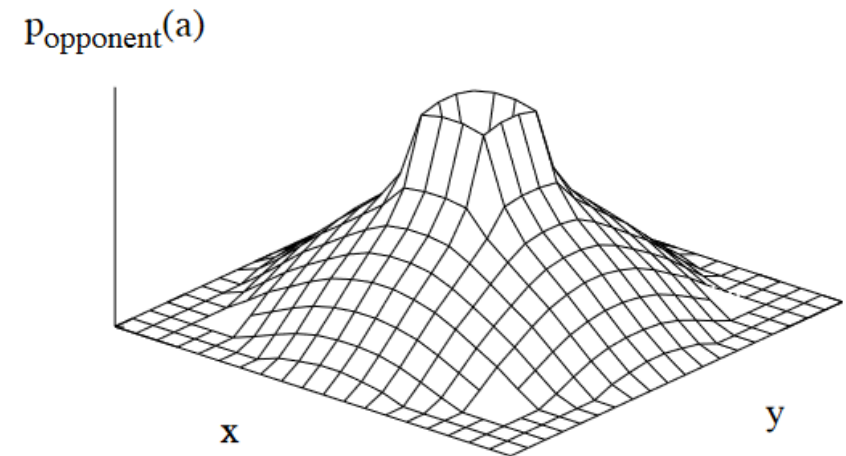
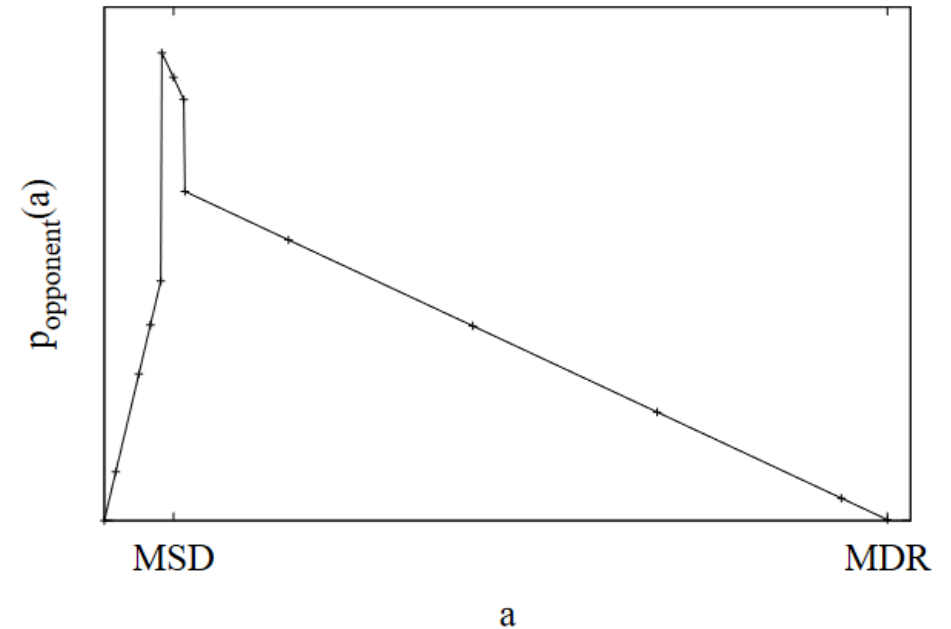




Undvik komma för nära

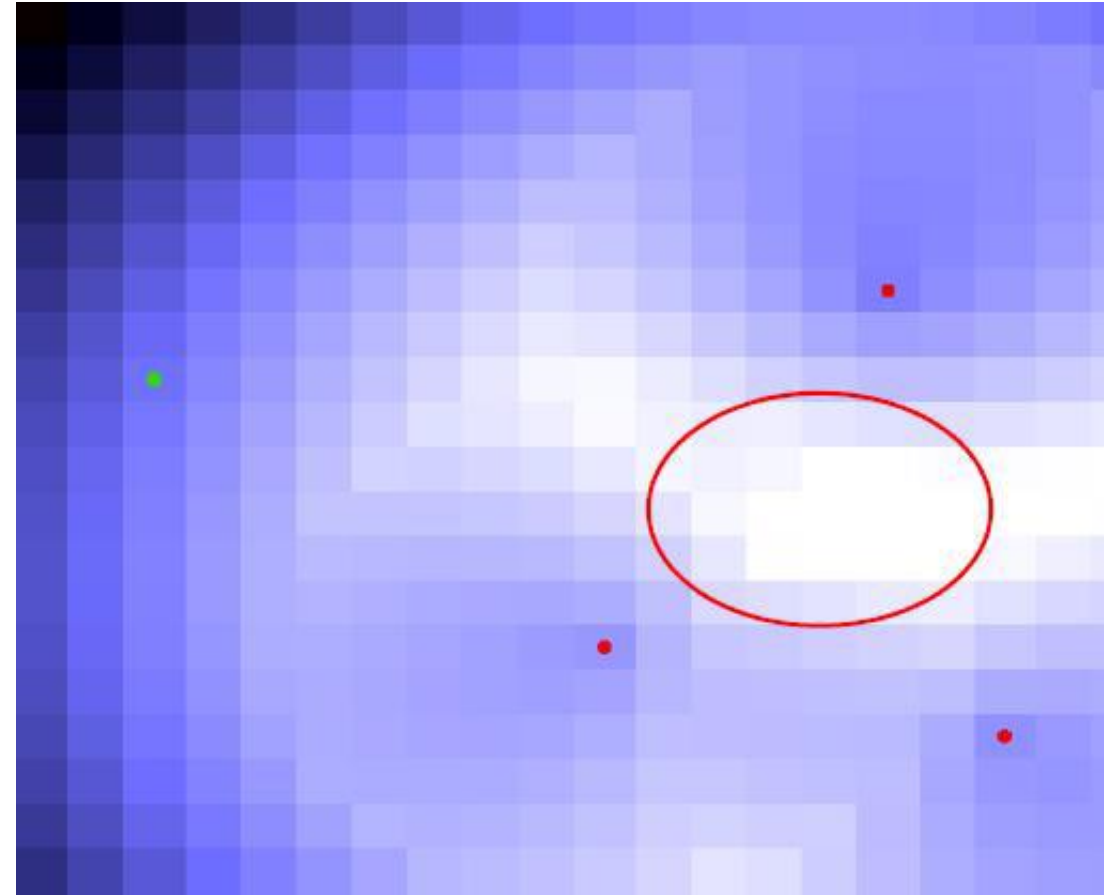
Potentialfält exempel:

- Dåligt (onödigt) att vara närmare fienden än MSD, *Maximum Shooting Distance*.
- Bra att vara precis vid MSD.
- Utanför *Maximum Detection Range* påverkas man inte (man ser inte fienden).



Undvik komma för nära

- Addera potentialer från olika attraherande och repellerande positioner.
- I varje steg, gå till den bästa potentialen inom n celler från där du är.





Utanför räckvidd



Mikromanagement

Teknikexempel: Förstärkningsinläarning med *Q-learning*

För varje beslut får man belöning/straff; vilken strategi blir bäst i längden?



Mikromanagement

Teknikexempel: Förstärkningsinlärning med *Q-learning*

1. Välj ut intressanta features.
Exempel: Egen hälsa, motståndarhälsa, relativt avstånd, stridskraft, ...
2. Diskretisera features.
Exempel: $\#h\ddot{a}lsa_a \in \{0, 10, 20, 30, \dots, 90, 100\}$ (procent)
3. Välj nyttofunktion (*utility function*) att maximera.
Exempel: $100 * \#trupper_a + \#h\ddot{a}lsa_a - 100 * \#trupper_{-a} - \#h\ddot{a}lsa_{-a}$
4. Konstruera representativa scenarier och träna fram Q-tabellen:
 $\Pr(handling \mid tillst\ddot{a}nd)$

Problem: Välja strategi och taktik

Exempel: Vi vill välja en **övergripande strategi** för vår agent

- Ska agenten vara Expansiv (bygga upp ekonomi och industri), Defensiv , eller Offensiv just nu?

Och/eller: Vi vill välja **taktik**

- Hur ska vi genomföra den valda strategin?

Problem: Välja strategi och taktik

Teknikexempel: *Supervised learning* med bayesianska nätverk

- $strategi \in \{Expansiv, Defensiv, Offensiv\}$
- $tillstånd \in \{\#baser, \#byggnader(x), \#enheter(y), \dots\}$
- Observationer: iakttagelser av motståndarens tillstånd (vad är intressant?)

$$\Pr(strategi_a \mid observationer, tillstånd_a) = ?$$
$$\Pr(strategi_{-a} \mid observationer) = ?$$

Vad är motståndarens strategi?



Vad är motståndarens strategi?



Tilldelningsproblemet



Agenter med olika förmågor

Tilldelningsproblemet



Agenter med olika förmågor

Tilldelningsproblemet

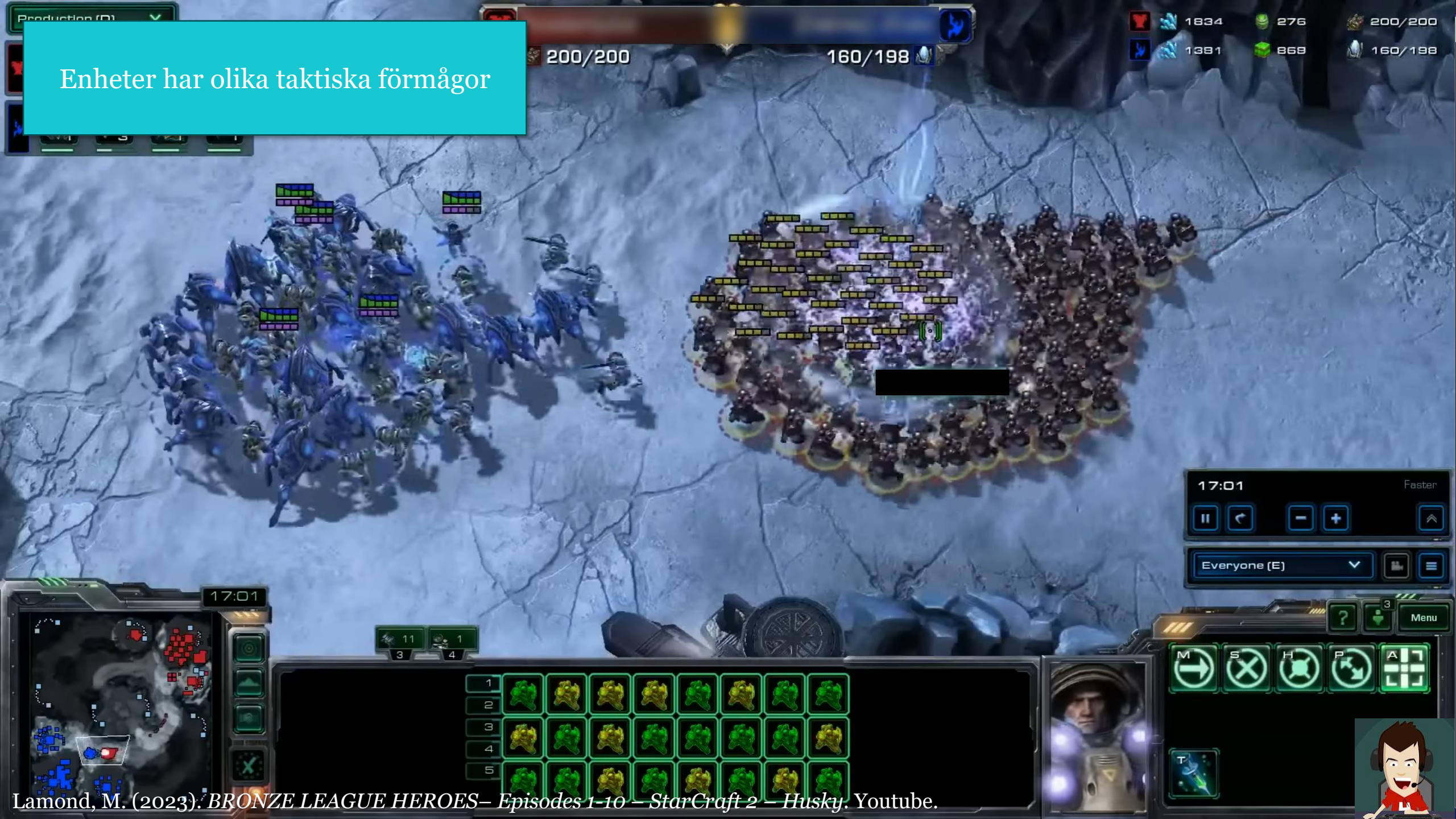
Kan, i *StarCraft II*, också användas till:

- **Mikromanagement**: Tilldela mål i strid, fokusera eldkraft.
- **Produktion**: Bestämma vilken byggnad som skall producera vad.
- **"Strategi"**: Tilldela abstrakta uppgifter (t.ex. "anfall bas" eller "försvara område") till grupper av enheter.

...och mycket mer!



Enheter har olika taktiska förmågor



Enheter har olika taktiska förmågor



Enheter har olika taktiska förmågor



Enheter har olika taktiska förmågor



Grupperingsproblemet



Agenter med olika förmågor

Grupperingsproblemet



Värde: 3



Värde: 1



Värde: 5

Totalt värde:
 $3 + 1 + 5 = 9$

Grupperingsproblemet

Teknik: *Coalition structure generation, teamgruppering, optimering, branch and bound*

StarCraft-exempel: Skapa effektiva team/grupper som har goda synergier (t.ex. *Medivac* som kan hela trupper + trupper som kan strida).

- $A = \left\{ \begin{array}{l} \text{Marauder}_1, \text{Marauder}_2, \text{Medivac}_1, \text{Medivac}_2, \\ \text{marinsoldat}_1, \text{marinsoldat}_2, \text{marinsoldat}_3, \text{marinsoldat}_4, \\ \text{marinsoldat}_5, \text{stridsvagn}_1 \end{array} \right\}$
- *Constraint:* Varje enhet måste vara medlem i exakt ett team.
- *Värdefunktion:* Många olika sätt att definiera...



Allokering av enheter till en ny bas



Nästa steg

Möjliga AI-problem

För att spela StarCraft II behöver agenten kunna...

1. Hämta in resurser på ett effektivt sätt
2. Skicka enheter för att observera motståndaren
3. Styra dessa enheter så att de inte dör för tidigt
4. Välja strategi beroende på vad motståndaren gör
5. Placera byggnader på strategiska platser, t.ex. flaskhalsar
6. Välja vilka enheter behöver byggas och när
7. Bygga dessa enheter på ett effektivt sätt
8. Allokera grupper enheter till uppgifter
9. Förflytta trupper i strid för att maximera/minimera skada

Exempel: Skelman-agenten



Utredningsplan

Glöm inte **utredningsplanen**:

1. Fundera över vad ni vill göra, ta inspiration från exemplen idag och **litteraturlistan** (Wiki).
2. Anmäla projektgrupper.
3. Kolla i projektgruppsförband att det **inte** finns överlapp.
4. Skriv och lämna in en individuell utredningsplan (=steg 2).

Utredningsplan för Liu Idsson (liuid123)

Allmän information om mallen

Detta är en mall för att skriva en utredningsplan i TDDD92-2023. Mallen beskriver de olika avsnitt som ska finnas och hur man ska fylla i dem. Normal storlek är omkring 1 sida.

Vi har fyllt på mallen med det vi ser att ni som kursdeltagare kan behöva tänka extra på. Vi hoppas att detta ska vara till hjälp och inte låter alltför överväldigande. Meningen är helt enkelt att ni ska få en rimlig chans att veta vad vi är ute efter och åt vilket håll ni ska sikta, så att ni förstår vad vi menar med tex. "problemställning" – inte att allt måste vara fullständigt perfekt från början.

De numererade rubrikerna nedan ska vara kvar och ska inte ändras, men alla instruktioner (inklusive denna och "allmän info"-rubriken) ska så klart tas bort innan inlämningen.

För att undvika att skriva en alltför luddig utredningsplan är det bra om du tänker dig att du beskriver ett arbetsområde för *någon annan*, som är den som faktiskt ska genomföra arbetet utan att behöva kontakta dig för att reda ut detaljerna!

På <https://www.ida.liu.se/~TDDD92/2023/utredning.shtml> får du mer information kring utredningen.

1 Problemställning

Här ska du beskriva och tydligt definiera ett **konkret och avgränsat problem som du vill lösa** i StarCraft. Vi förtydligar:

1. Du ska beskriva ett **problem i StarCraft**. Ursprungligen kanske du var mer intresserad av att testa en viss AI-teknik och letade efter ett problem du kunde lösa med detta. Här måste du ändå beskriva det problem du hittade: Du vill se till att agenten kan göra *något*, åstadkomma *något*, bestämma *något*, ha en viss *förmåga*... för att den sedan ska kunna spela bättre.

Att "jämföra Q-learning med supervised learning" är inte något som direkt uppkommer från StarCraft utan handlar om att undersöka egenskaper hos tekniker. Att göra "opponent modeling" (punkt) är inte tillräckligt specifikt och konkret. Hitta istället ett problem vars lösning direkt bidrar till att spela bättre StarCraft.

2. Problemet ska vara **konkret och avgränsat** med en tydlig definition.

På hög nivå har man kanske en vision att "agenten ska vinna oftare" eller att "marines ska överleva längre" – men det ligger på en så hög nivå och är så brett i sin definition att det inte ger dig någon särskild vägledning, och då blir det svårt för dig själv att styra din utredning åt rätt håll.

Du behöver beskriva problemet så tydligt att läsaren förstår *precis vad du vill uppnå*. För att nå dit kan du tänka dig att du i själva verket **skriver en kravspecifikation som ska lämnas över till någon annan**. Vad behöver du skriva för att den personen ska arbeta på rätt uppgift fram till du kommer tillbaka om ett par månader? Dina tankar får inte stanna kvar i huvudet utan behöver komma ner på papper.

Här är några viktiga steg i att **vara tydlig**:

- Definera dina termer och begrepp noggrant. Vad menar du med orden?
- Tala om i vilka StarCraft-situationer problemet behöver lösas. Vad gör/har motståndaren, vad gör/har du, vad händer i spelet när ditt problem är aktuellt?
- Tala om vilka antaganden du gör om världen.

Lösningen på ditt valda problem kommer att behöva implementeras och integreras i en agent, där din implementation ska interagera med omvärlden. Du behöver *indata* och genererar *utdata*; du blir *anropad* och du kanske *anropar andra*. Du behöver beskriva hur detta är tänkt, både för tydlighetens skull och för att kunna samordna med dina gruppmedlemmar.

- **Hur anropas funktionaliteten?** Vill du att den ständigt kör "i bakgrunden" och gör något, tex. styr scouting? Eller tänker du dig att någon anropar dig när du ska göra eller beräkna något? Skickar de med några parametrar då? Vilka?

- **Vilken information förutsätter du finns tillgänglig för dig?** Det kan gälla information om världen/kartan, om motståndaren, om vad andra delar av gruppens agent håller på med, med mera. Definiera den informationen tydligt, så att man kan avgöra om informationen räcker till för att faktiskt lösa problemet!

Exempel: *Inom klassisk planering förutsätts tillgång till (a) information om exakt vilka handlingar som finns och deras precisa exekveringsvillkor (preconditions) och effekter, (b) fullständig information om världens nuvarande tillstånd (vilka fakta som är sanna och vilka som är falska), och (c) ett målvillkor uttryckt i termer av vilka fakta som ska bli sanna efter att en lösningsplan har exekverats.*

Tänk dig att den information som inte tas med här inte kommer att finnas tillgänglig vid körningen!

- **Vilken information tänker du dig att ge till resten av systemet?** Även här behöver du vara tillräckligt tydlig

Sammanfattning

Sammanfattning

Tänk på:

- ... att en lösningsteknik kräver **indata**
 - Sökmeter kräver en graf. Vad innehåller den? Var kommer den ifrån?
 - Potentialfält kräver attraktorer / repellerare. Var kommer de ifrån?
- ...och ger **utdata**, eller **styr** något
 - Potentialfält ger ett vektorfält. Hur använder man detta?

Sammanfattning

Tänk på:

- ... att det finns en **litteraturlista** med relevanta referenser på kursens **Wiki-sida**.
- ... att ni **inte** måste välja något av de problem som vi exemplifierade, eller lösa dem med de tekniker vi tog upp.
- ... vad er agent får för problem om ni saknar en bra lösning.

Sammanfattning

Det finns många intressanta AI problem att lösa och många tekniker och algoritmer att experimentera med!

Fundera på vad ni vill göra! Komplettera varandra!

Nästa föreläsning: FÖ3 Rapportskrivning (IKOS)