

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 t.ex. “problemställning” – inte att allt måste vara fullständigt perfekt från början.

De numrerade 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 noggrannt. 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, t.ex. 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 finns tillgänglig vid körningen!*

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

och konkret, inte minst för att man ska kunna avgöra vilka lösningsmetoder som är lämpliga. Om du t.ex. vill välja strategi, tala då om vilka alternativa strategier du har tänkt dig (du kan så klart ändra dig i det senare utredningsarbetet).

Exempel: *Inom klassisk planering är utdata en sekvens av utvalda handlingar sådan att om världen är i det angivna tillståndet, och sekvensen exekveras i den angivna sekvensen, kommer målvillkoret att bli sant i det resulterande tillståndet.*

Eller tänker du dig att din funktionalitet inte ger information, utan istället styr något? Beskriv då detta. Exempelvis kan det finnas en funktionalitet som direkt styr hur agenter rör sig vid en attack.

Du behöver också tänka på följande punkter:

- **Problemet måste avgränsas väl**, så det syns vad du vill försöka lösa men också vad du *inte* tänker lösa. Du får mycket gärna uttryckligen skriva att vissa problemställningar *inte* ingår!

- **Problemet behöver ha lagom omfattning / djup.**

Tänk på att utredningen är på 2 hp och att du inte ska försöka lösa för många problem på en gång. Då hinner du inte gå på djupet eller hinner inte alls att göra din utredning färdig. Däremot kan du mycket väl utreda flera olika *lösningar* på ditt problem.

Samtidigt får problemet inte vara alldeles för enkelt. Det är trivalt att ta reda på vilka *byggnadstyper* man måste konstruera innan man kan bygga något av typ X: Det kan lätt hårdkodas och lösas annars genom att titta på Tech Tree med en trivial algoritm. Du behöver få en chans att visa upp att du kan applicera AI-tekniker!

- **Djupet i utredningen kan komma från flera håll.**

Ett sätt att få ett djup i utredningen är nämligen att *algoritmerna* som krävs är lite mer komplexa. Ett annat är att du visserligen kan använda enklare algoritmer men behöver ett större djup i *modelleringen*.

Att räkna ut sannolikheter från ett Bayesianskt nätverk kan t.ex. vara enkelt, men det finns desto mer djup i modelleringsarbetet där du måste lägga ner arbete på att undersöka vilka noder och kopplingar som ska finnas i detta nätverk och på att komma fram till bra sannolikhetsstabeller. Arbetar man med ungerska metoden kan man behöva fundera mycket på hur man definierar tasks och hur man ska definiera sin utilityfunktion. Där finns ett djup i modelleringen.

Men om man applicerar djupet-först-sökning på Tech Tree har man *både* en enkel algoritm och en enkel modell, vilket blir problematiskt.

2 Tekniker och algoritmer

Nu vet du vilket problem du vill lösa, och vi har på annan plats angivit ett antal generella kriterier som vi tycker att en bra lösning ska uppfylla. Då kan du gå vidare och titta på vilka specifika lösningsmetoder som *kan tänkas* vara användbara för att lösa problemet, och välja ut en eller flera som ska undersökas närmare.

Allmänt om lösningsmetoder: Det du ska ha tillgängligt för att lösa problemet du ställde upp är (1) de indata som du tänker dig att få från omvärlden och från resten av agenten, och (2) en existerande *AI-algoritm*, som du kan behöva implementera själv.

Algoritmen tar inte nödvändigtvis exakt samma indata som de du själv har tillgängliga. Många *automatiska planerare* vill till exempel ha en problembeskrivning i språket PDDL, vilket bland annat kräver att du skapar en *världsmodell* i det formatet (se även labbar i TDDC17). Vill du använda en sådan planerare, men indata är i form av ett Tech Tree och en lista på tillgängliga resurser, behöver du ett sätt att transformera dina indata till ett PDDL-problem. Även detta ingår i en lösningsmetod.

I ett annat fall kanske du har vissa sannolikheter tillgängliga, men behöver skapa ett Bayesianskt nätverk för att kunna applicera vissa algoritmer.

Utredningen kommer att behöva gå genom alla dessa delar av en lösning. Här i utredningsplanen är vi *mest* intresserade av de *algoritmer* som du tänker utreda, men för din egen skull är det bra om du också tänker lite på hur du kan få tag på de data som algoritmerna vill ha – så du inte upptäcker långt senare att detta är omöjligt. Det *kan* också tänkas att du kan applicera samma algoritm på två helt olika sätt (olika modeller), och därmed få två olika lösningsmetoder!

En lösningsmetod eller flera? Det *kan* fungera att göra en djupdykning i *en* teknik, men då bör du vara relativt säker på att den faktiskt fungerar för problemet. Annars kan du få problem i slutet när du bestämmer dig för att tekniken *inte* fungerar och behöver hitta en ny. För att utvärderingen ska vara tillräckligt djup behöver du då också fokusera extra mycket på att utvärdera *hur* bra lösningen uppfyller de olika kriterierna.

Det finns annars fördelar med att jämföra *flera* olika lösningar. Till exempel kan det vara lättare att komma fram till att *lösning A är bättre än lösning B på kriterium X* än att komma fram till *exakt hur bra lösning A är på kriterium X*.

Kan lösningsmetoder vara för lika varandra? Ja, i vissa fall kan det hända, speciellt med tanke på att man behöver kunna utvärdera dem *utan att implementera dem*. Att jämföra två enklare sökalgoritmer som DFS och BFS kan t.ex. lätt ge ett alltför simpelt resultat: Man använder troligen samma sökrymd, båda är triviala att implementera, och det kan lätt bli uppenbart att vissa algoritmer inte alls passar in.

Det kan också vara orimligt att jämföra oinformerad sökning med informerad sökning; om man har någon information att använda, är informerad sökning alltid att föredra.

Generella eller specifika metoder? Lösningsmetoder ska normalt vara generellt applicerbara. Även om vi t.ex. enbart använder en enda karta i kursen, ska man inte använda tekniker som man bara fungerar om man hårdkodar en specifik karta, eller som kräver väldigt specifika egenskaper hos en karta.

Samtidigt måste metoderna vara konkreta. *Reinforcement learning* är exempelvis en *kategori* av metoder som har vissa gemensamma egenskaper på en abstrakt nivå. Detta är ingen lösningsmetod utan behöver konkretiseras. Även *supervised learning* och *unsupervised learning* är kategorier av metoder, liksom *sökning* och *automatisk planering*. Däremot är

Q-learning en konkret metod inom kategorin reinforcement learning.

Hur väljer man lösningsmetoder att utreda?

- Gå genom litteraturen – artiklar som vi har föreslagit och gärna andra alternativ som du hittar själv.
- Skumma genom dem i upp till 20–30 minuter per artikel.
- Bilda en uppfattning om vilka metoder du vill titta närmare på under utredningen.

Det går sällan att göra en bra utredning utifrån en enda artikel. Vi förutsätter att du hittar *flera källor* att använda i utredningen, även om du bara undersöker en lösningsmetod.

Artiklar baserade på StarCraft II? I litteraturen finns en hel del artiklar som applicerar olika AI-tekniker på just StarCraft II, och som redan utvärderar resultatet av detta.

Om en individuell utredning enbart baserar sig på *en* teknik av denna typ, försvinner mycket av det tänkta utredningsarbetet. Rapporten säger då i princip att *Jag vill tackla problemet X inom StarCraft II och vill reda ut om teknik Y skulle vara användbar enligt kriterierna Z1, Z2 och Z3. John Doe [7] har applicerat teknik Y på problemet X inom StarCraft II och kommit fram till att kriterierna Z1 och Z2 uppfylls bra medan Z3 uppfylls mindre bra. Slut på utredningen.*

Ovanstående är så klart något överdrivet, men det är ändå lätt att hamna i denna fälla och att därmed inte komma fram till något rimligt djup i den egna utredningen. Om du baserar din utredning på existerande vetenskapliga artiklar i StarCraft-miljö behöver du därför:

- Utvärdera *flera* olika tekniker och göra en egen slutlig jämförelse mellan dem (t.ex. använda flera tekniker som redan applicerats på StarCraft, eller en som redan applicerats och en som inte har gjort det), eller
- Utvärdera *en* teknik som redan har applicerats på StarCraft, men komma fram till egna *djupare slutsatser* som inte framgår i ursprungsartikeln/artiklarna (vilket kan vara svårt att nå upp till). Diskutera i så fall detta i förväg, med grund i dina tänkta val av specifika artiklar.

Så vad ska avsnittet innehålla egentligen? Du behöver ange vilka tekniker/lösningar du har tänkt undersöka närmare i utredningen. Ange tydligt om du har tänkt dig att vissa tekniker är *alternativ* eller *komplement* till varandra.

Fyll i en numrerad lista på följande sätt, där varje punkt är ett tänkbart *alternativ*. Om du bara utvärderar ett alternativ – är detta bra eller dåligt? – blir det alltså bara en punkt (nummer 1) i listan. Låt ändå din information vara kvar i listform så din punkt behåller sitt nummer!

1. Teknik/lösning A tillsammans med teknik/lösning B.
2. Teknik/lösning C, ensam.
3. Teknik/lösning D tillsammans med teknik/lösning B.

I varje punkt ska du också inkludera *vetenskapliga referenser* till dina lösningar. Wikipedia kan fungera som komplement men är inte en vetenskaplig referens. Ett separat dokument beskriver hur man hanterar referenser med BibTeX.

Riskabla tekniker. Slutligen vill vi varna för att vissa typer av tekniker har sina risker – de kan mycket väl fungera alldeles utmärkt, men vi har sett att några (inte alla) av de som satsar på dessa tekniker behöver lägga ner mycket arbete under den senare projektdelen av kursen.

Om du väljer att *utvärdera* sådana tekniker kan det vara bra att ta med alternativa tekniker i utvärderingen. Då kan du senare, när du har läst på och vet mera, ta ett välgrundat beslut: Vilken av dina utvärderade tekniker vill du *gå vidare med* under projektet? Det är ju detta som är meningen med utvärderingen!

- Tekniker som baseras på **maskininlärning** är intressanta för många. Samtidigt blir resultatet alltid beroende av hur väl inlärningen lyckas, och detta är något som man inte kan påverka till 100% själv. I media ser vi bara de lyckade exemplen.

Man kan så klart bli godkänd även om utfallet (i fråga om hur bra agenten spelar) inte blir det man har önskat sig, men man kan då behöva lägga ner mer arbete på att förklara varför det hände och att visa att den inte helt enkelt är den grundläggande implementationen som är fel.

Om du använder maskininlärning, försök då bli klar med implementationen i god tid. Det räcker inte med att köra själva inlärningen en gång, utan du kommer med största sannolikhet att vilja fixa buggar, köra om, prova att justera någon parameter, köra om... och varje ny inlärningsomgång skulle kunna ta flera dagar.

- **Deep Learning** syns speciellt mycket i media och kräver särskilt mycket tid, både datortid och tid att förstå vad man ska göra. Vi rekommenderar *mycket* starkt att man inte ger sig in i det området i den här kursen.
- Användandet av **replays / repris**er kan ta lång tid. Det kan också vara svårt att hitta tillräckligt många / tillräckligt tillämpliga reprisar som kan läsas i verktygen och som har den information som du behöver.
- Tänk på att **supervised learning** kräver att man har en form av “facit”. Vill du lära dig i vilka situationer man bör spela offensivt respektive defensivt? Du kan säkert hitta gott om intressanta *indata* i replays – hur många enheter du har, hur många enheter motståndaren har, med mera – men det är desto svårare att automatiskt analysera om spelaren *spelade offensivt eller defensivt* vid en specifik punkt i en specifik replay. Då blir det upp till dig att manuellt skapa sådana “etiketter/labels” för en mindre uppsättning replays.
- Vi har främst använt en och samma **karta** under kursen. Det finns en karteditor, men att använda den kan vara mer knepigt.

3 Övrigt

Här kan du – om du vill – ta upp annat som du tror kan vara intressant för att andra ska förstå vad du tänker utreda.

Innan inlämning

Har du gjort ditt bästa för att specificera problemet, så att en *annan* person skulle kunna förstå din beskrivning tillräckligt bra för att göra hela utredningen?