|  |
| --- |
|  |
| Laboration 2 |
| Rapport |
|  |
|  |
|  |
|  |

**Innehåll**

[1 Instruktioner - 1 -](#_Toc436247938)

[2 BridgeGrid (Value Iteration) - 1 -](#_Toc436247939)

[3 DiscountGrid (Value Iteration) - 2 -](#_Toc436247940)

[4 BridgeGrid (Q-Learning) - 2 -](#_Toc436247941)

[5 Optimering med den Genetiska Algoritmen - 3 -](#_Toc436247942)

# Instruktioner

Denna rapporten hör till laboration 2 (MDPer, RL och GA). I laborationsdokumnetet ombeds ni fylla i ett antal uppgifter i denna rapporten. När ni lämnar in laborationen skall denna ifyllda rapporten finnas med i arkivfilen.

# BridgeGrid (Value Iteration)

Byt ut värdet för antingen *discount* **eller** *noise* parametern i nedanstående tabell. Med det modifierade värdet skall value iteration agenten kunna följa den genererade policyn för att korsa bron och lämna Bridgeworld brädet via terminaltillståndet +10.00. Förklara också varför parameterinställning leder till önskad policy.

|  |  |  |
| --- | --- | --- |
| **Parameter** | **Värde** | **Förklaring** |
| discount | 0.9 | Ett noise värde på noll leder till att agenten aldrig riskerar att hamna någonstans som den inte planerat att hamna på, detta är bra i detta exempel eftersom att varje steg som inte är åt höger eller vänster resulterar i ett terminaltillstånd med negativt beting.  Discount värdet 0,9 innebär att en belöning som är långt borta kommer vara värt relativt mycket även att det är långt borta. |
| noise | 0.0 |
|  | |

Uppdateringsformeln för value iteration återges nedan

och är ekvivalent med

där

# DiscountGrid (Value Iteration)

Fyll i de tre parametervärdena *discount*, *noise* och *livingReward* (*reward*) som leder till önskad policy och förklara även varför dessa parameterinställningar leder till önskad policy.

|  |  |  |  |
| --- | --- | --- | --- |
| **Uppgift** | **Parameter** | **Värde** | **Förklaring** |
| a | discount | 0.4 | Discount 0,4 gör att det är mer värt att gå till det mindre värda terminaltillståndet för att det är närmre starttillståndet. Ett negativt reward värde gör att agenten tycker det är ”unpleasant” att befinna sig i den aktuella världen då den ju längre tid den gör detta ju mer negativ poängsumma får den och vill därför ta en så kort väg till ett terminaltillstånd som möjligt. Ett noise värde på noll gör agentens handlingar deterministiska och kommer därav ej riskera att hamna någon annan stans än dit agenten avsett att hamna. |
| noise | 0.0 |
| reward | -0.5 |
|  | |
| b | discount | 0.2 | Discount 0,2 gör det mer värt att den går till det mindre värda terminaltillståndet. noise 0,2 gör att den inte tar risken att hamna i ett negativt terminaltillstånd. Då agentens noise värde är 0 så gör detta att agenten är mindre benägen till att ta onödiga risker, som exempelvis att gå förbi de negativa terminaltillstånden då det finns en sannolikhet att agenten kan hamna i dessa fastän den inte planerat detta. Detta på grund utav att agentens handlingar är stokastiska då noise parametern är > 0. |
| noise | 0.2 |
| reward | 0.0 |
|  | |
| c | discount | 0.9 | Discount 0,9 gör det mer värt att välja det terminaltilltånd med högre värde. noise 0,0 gör det riskfritt att gå den korta vägen. En negativ living reward gör agenten gör att agenten tycker att miljön är ganska ”unpleasant” och av den anledningen vill nå ett terminaltillstånd så snabbt som möjligt. Då det inte finns någon risk att den hamnar i ett negativt terminal tillstånd då noise == 0 så väljer agenten självklart att ta den snabbast möjliga vägen till terminaltillståndet. |
| noise | 0.0 |
| reward | -0.5 |
|  | |
| d | discount | 0.4 | Discount 0,4 gör att agenten föredrar det mer värda terminaltillståndet, och noise 0,2 gör att den tar den säkra omvägen istället för att riskera  terminaltillstånden med negativa värden. |
| noise | 0.2 |
| reward | 0.0 |
|  | |
| e | discount | 0.0 | 0,0 discount gör att agenten inte vill gå till något terminaltillstånd då det inte har något värde. Noise 0,0 gör att den inte råkar hamna i ett terminaltillstånd. Reward gör att den håller sig vid liv i ett hörn för att få mer poäng. |
| noise | 0.0 |
| reward | 1.0 |
|  | |

# BridgeGrid (Q-Learning)

Fyll i de två parametervärdena *epsilon* och *alpha* som leder till den optimala policyn under 50 träningsepisoder. Om den optimala policyn inte kan hittas med någon parameterkombination under 50 träningsepisoder, ange värdet *GÅR EJ* för de båda parametrarna. Motivera också erat svar, dvs varför hittas den optimala policyn med en viss parameterkombination eller varför går det inte att hitta den optimala policyn med någon parameterkombination?

|  |  |  |
| --- | --- | --- |
| **Parameter** | **Värde** | **Förklaring** |
| epsilon | GÅR EJ | *Har man ett högre epsilon värde så kan agenten hamna där på ren tur då detta gör att agenten utforskar mer. Men för att agenten ska kunna förstå att detta är den optimala policyn hade antagligen flertalet träningsepisoder behövts.* |
| alpha | GÅR EJ |
|  | |

# Optimering med den Genetiska Algoritmen

Fyll i de parametrar ni anser vara optimala för att hitta maximat för funktionerna 1-4.

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Function | #Gens | PopSize | #Genes | Pcross | Pmut | Selection Method | Ptour | Tournament  Size | #Elite |
| f1 | 100 | 10 | 50 | 0,8 | 0,1 | Tournament | 0,75 | 4 | 1 |
| f2 | 100 | 20 | 50 | 0,8 | 0,1 | Tournament | 0,75 | 4 | 1 |
| f3 | 150 | 100 | 80 | 0,8 | 0,1 | Tournament | 0,75 | 5 | 10 |
| f4 | 100 | 150 | 80 | 0,8 | 0,1 | Tournament | 0,75 | 6 | 4 |