TBMI26 – Computer Assignment Reports Reinforcement Learning

Deadline – March 15 2019

Author/-s:
Kristian Sikiric
Kaj Rolandsson

In order to pass the assignment you will need to answer the following questions and upload the document to LISAM. **You will also need to upload all code in .m-file format**. We will correct the reports continuously so feel free to send them as soon as possible. If you meet the deadline you will have the lab part of the course reported in LADOK together with the exam. If not, you'll get the lab part reported during the re-exam period.

1. **Define the V- and Q-function given an optimal policy. Use equations and describe what they represent. (See lectures/classes)**
   $Q(s_k,a) = r(s_k,a) + \gamma*V^*(s_{k+1})$
   $V^*(s) = \max(Q(s,a))$

   Q(s,a) is the expected future reward of doing action a in state a and the following the optimal policy.

   V*(s) is the maximum reward in state s, it is found by maximizing the Q function over the actions.

2. **Define a learning rule (equation) for the Q-function and describe how it works. (Theory, see lectures/classes)**

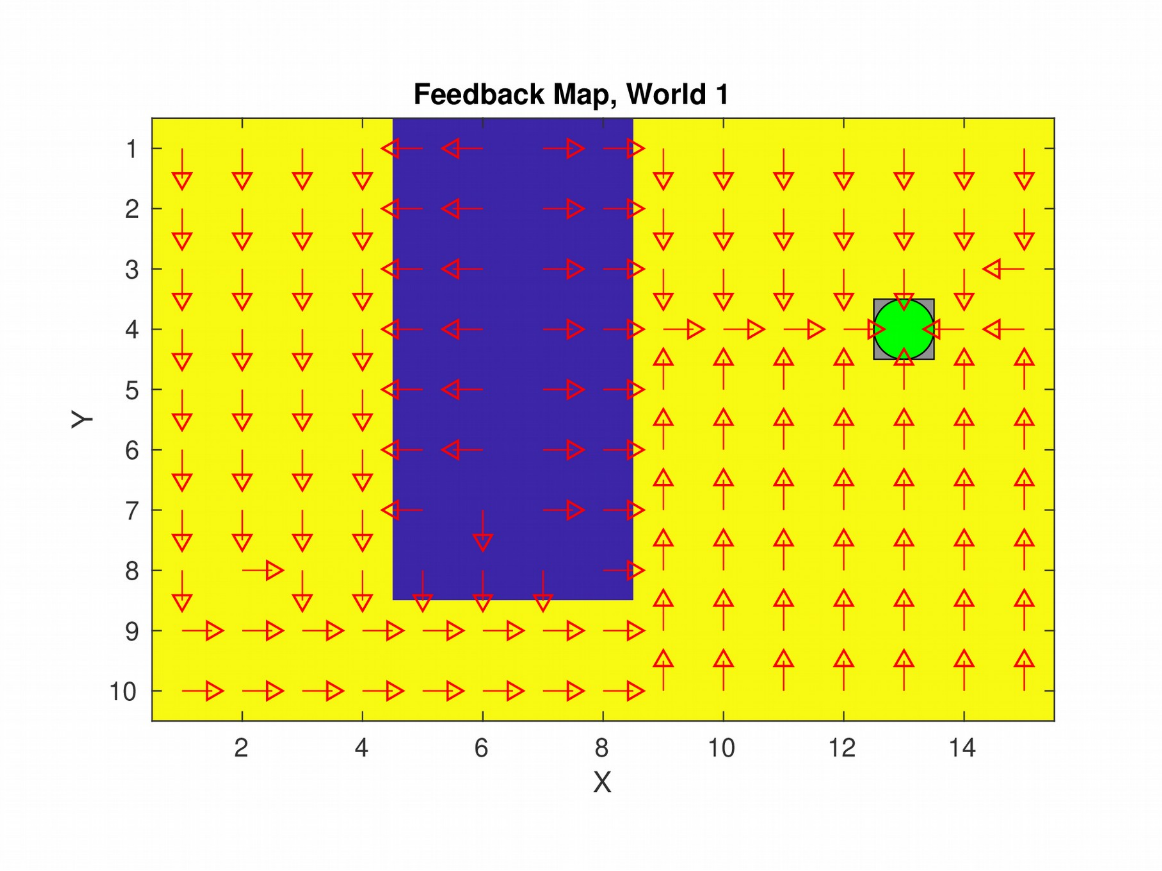
   $Q(s_k,a_j) = (1-\eta)*Q(s_k,a_j) + \eta(r+\gamma*\max(Q(_{sk+1},a)))$,

   where η is the learning rate, γ is the discount factor, s_k is the current state, a is the action chosen and s_{k+1} is the state after action a and r is the reward in the state after action a.

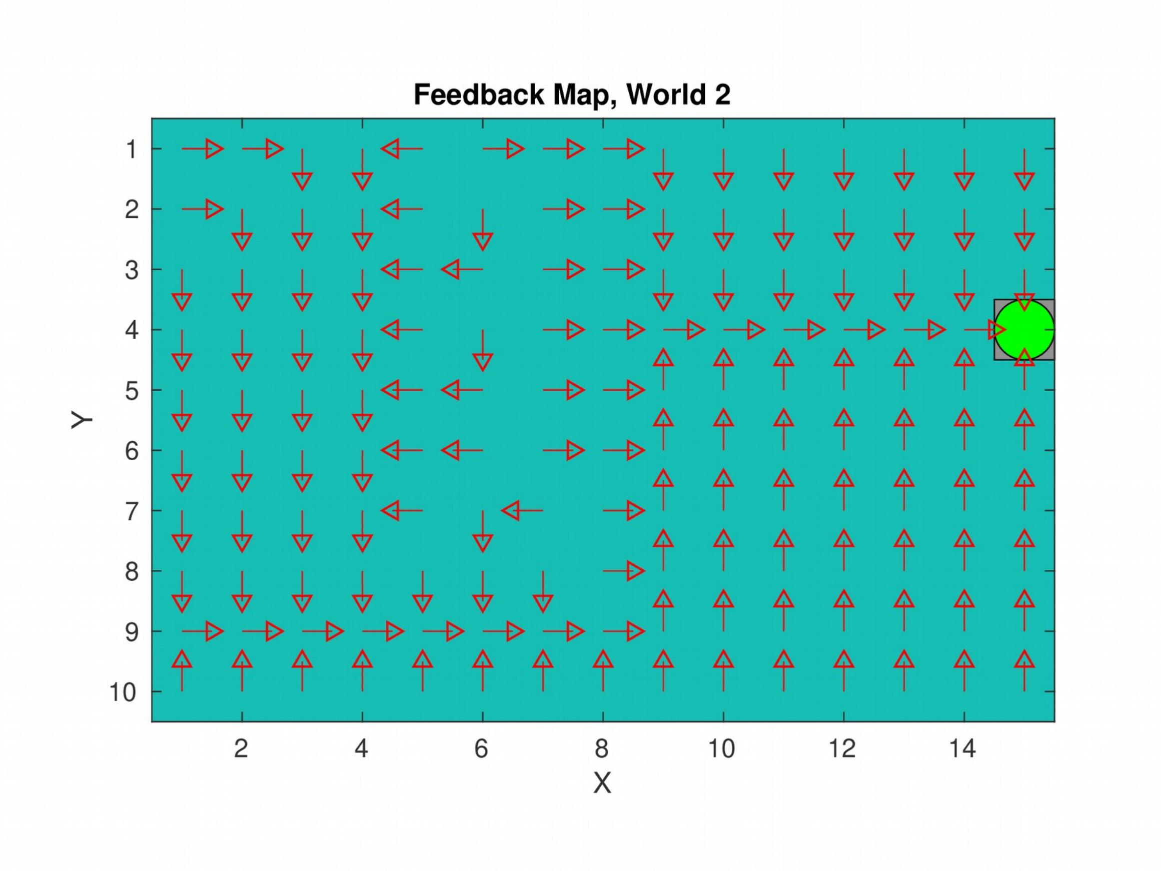
   This equation shows how the Q function changes with new information. The η variable specifies to which extent new information will be taken into account. The variable γ specifies how short or long sighted the updates are. Large γ results in a larger emphasis on long term rewards. The the Q value is calculated using the old Q value, the reward and the best possible value in the next state.

3. **Briefly describe your implementation, especially how you hinder the robot from exiting through the borders of a world.**
   First we initialize the Q table with random values between 0 and 1 and set the different parameters described above. Then we initialize a start state and update the Q function until the goal is found. If an action is invalid, we set the Q value of that state and action to minus infinity.

4. **Describe World 1. What is the goal of the reinforcement learning in this world? What parameters did you use to solve this world? Plot the policy and the V-function.**
   Static world with obstacle, the goal is to find shortest path from start to finish. Learning rate is set to 0.3, discount factor to 0.9 and the exploration rate to 0.9 for the first 40% of training, then it is set to 0.3.

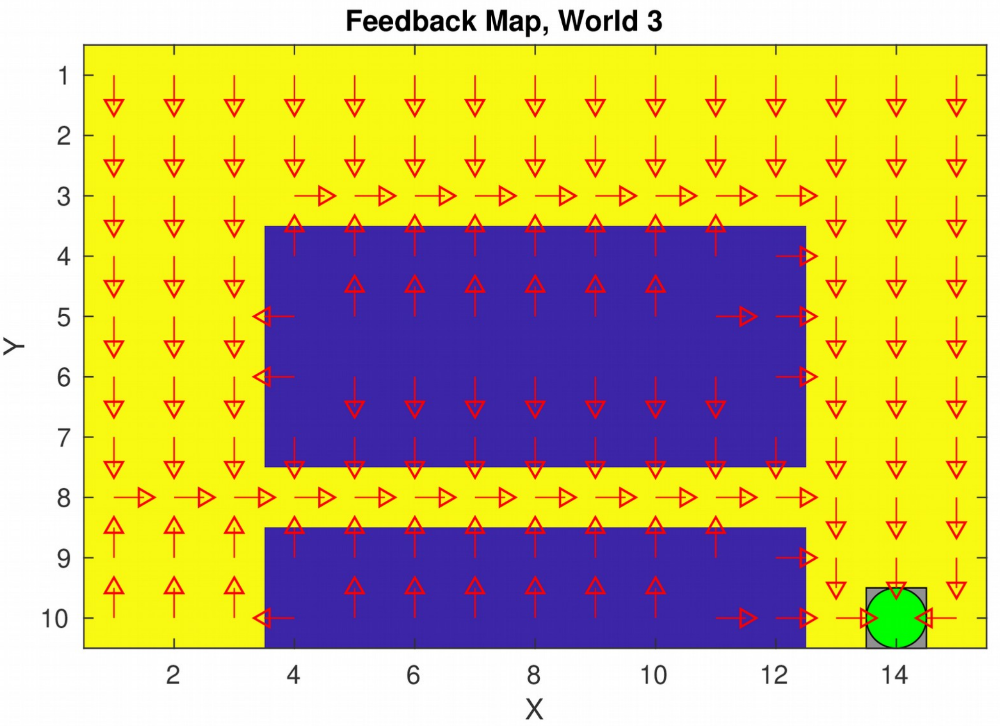


5. **Describe World 2. What is the goal of the reinforcement learning in this world? What parameters did you use to solve this world? Plot the policy and the V-function.**

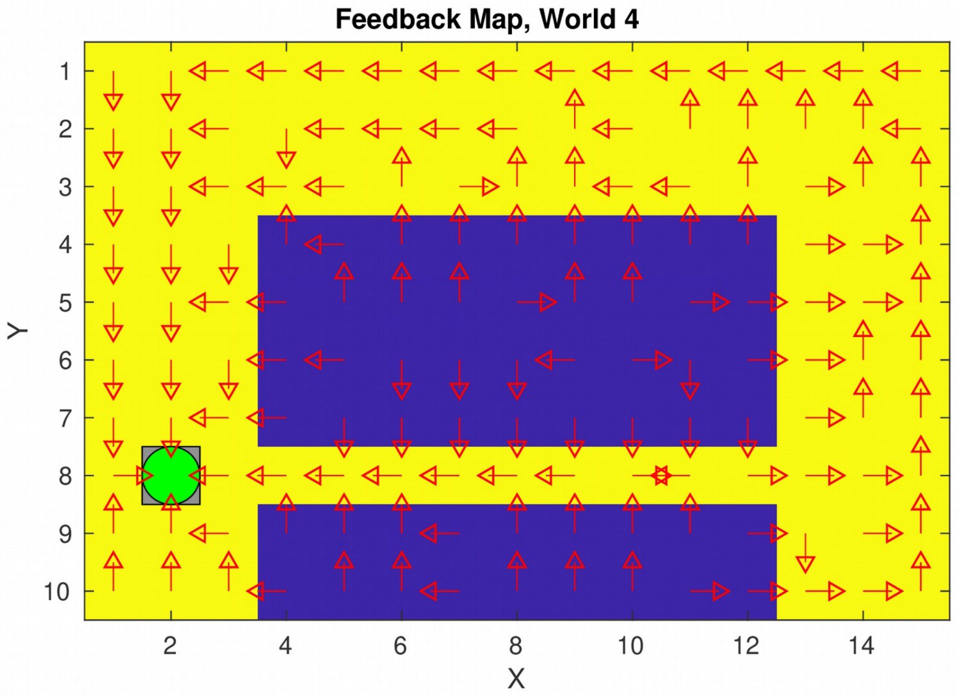


Sometimes there is an obstacle and sometimes not, the goal is to find the path that maximizes the reward to the goal as if the obstacle is present. Learning rate is set to 0.3, discount factor to 0.9 and the exploration rate to 0.9 for the first 40% of training, then it is set to 0.3.

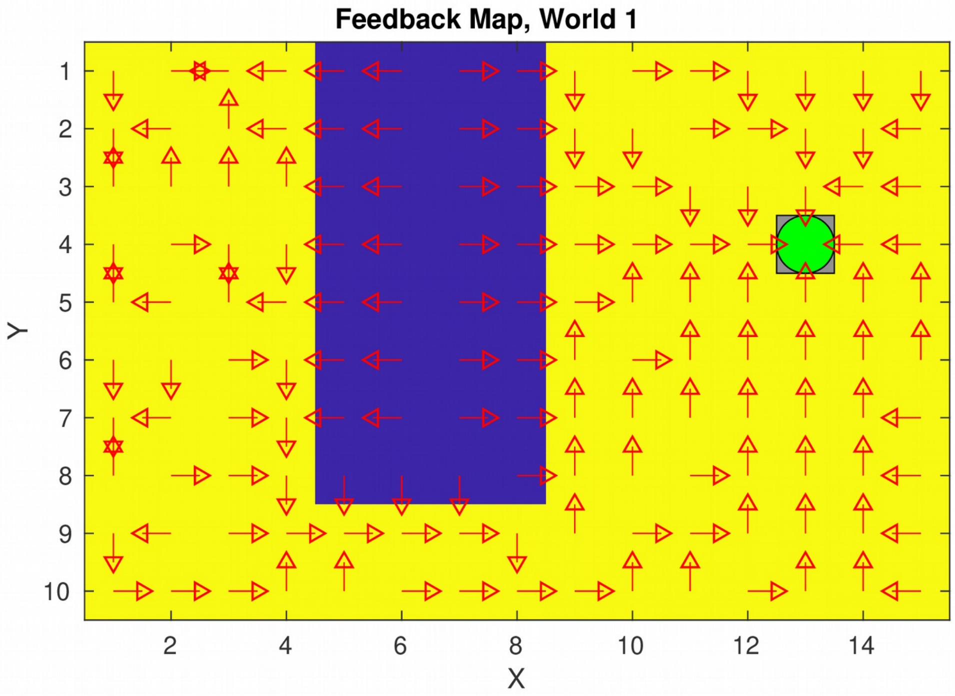
6. **Describe World 3. What is the goal of the reinforcement learning in this world? What parameters did you use to solve this world? Plot the policy and the V-function.**
   Static world, the goal is to find the path that maximizes the reward to the goal. Learning rate is set to 0.3, discount factor to 0.9 and the exploration rate to 0.9 for the first 40% of training, then it is set to 0.3.



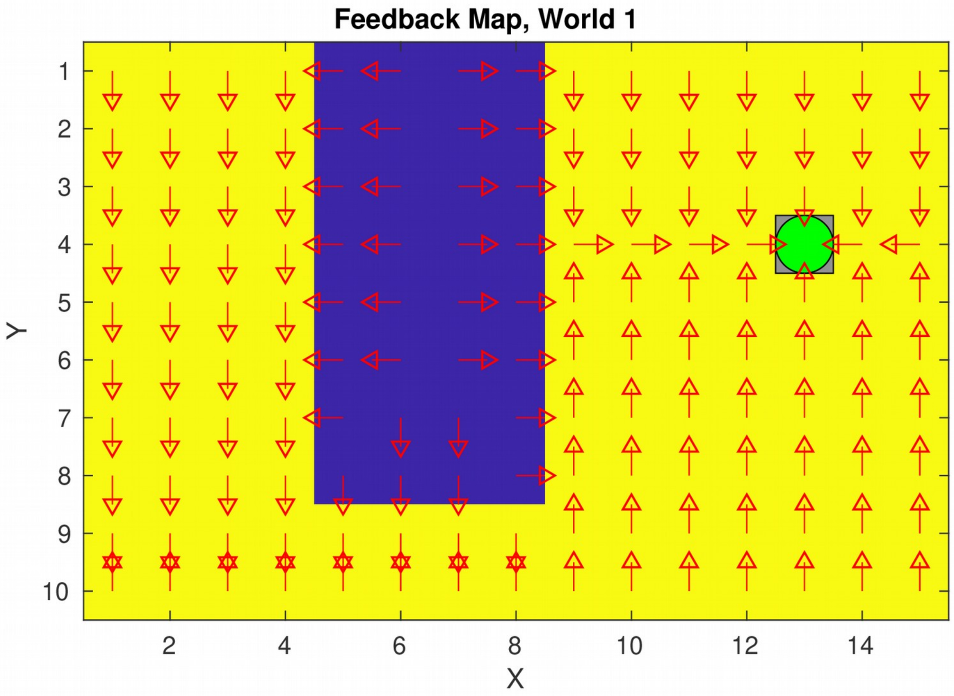
7. **Describe World 4. What is the goal of the reinforcement learning in this world? How is this world different from world 3, and why can this be solved using reinforcement learning? What parameters did you use to solve this world? Plot the policy and the V-function.**
   Same as world 3, but there is a 30% chance that a random action is taken. Since there is a risk of taking the wrong action, it is more “safe” to take the long path than the shortest (in this case). Learning rate is set to 0.3, discount factor to 0.9 and the exploration rate to 0.9 for the first 40% of training, then it is set to 0.3.



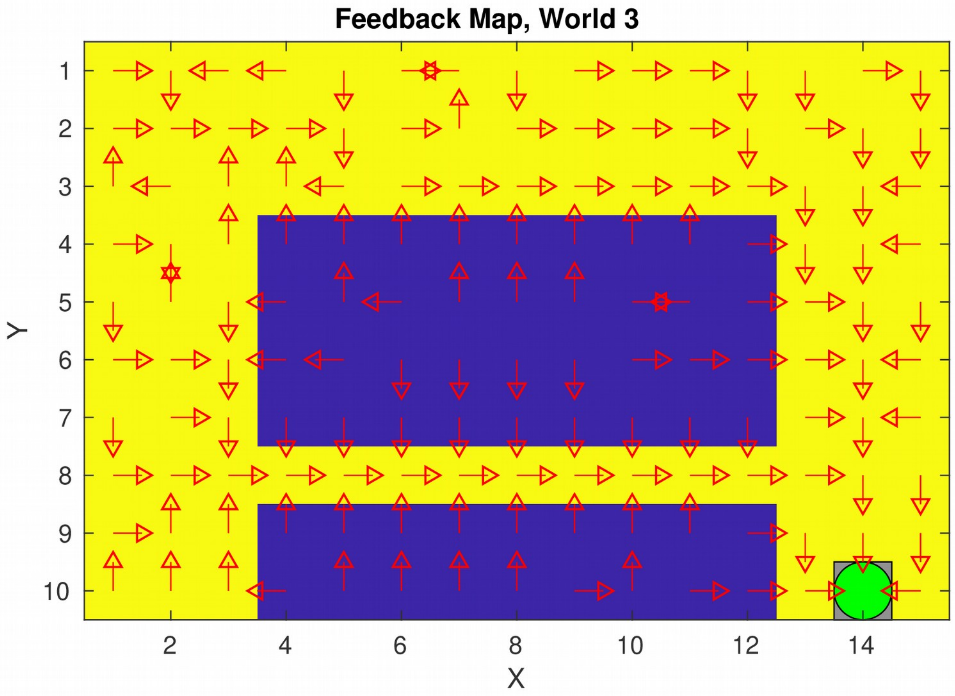
8. **Explain how the learning rate α influences the policy and V-function in each world. Use figures to make your point.**
The learning rate decides how much new values will be taken into account. Higher learning rate puts more emphasis on new value, while a lower learning rate does not care so much. But a too high learning rate might make the model diverge. In the case of Q-learning, a low learning rate will put more emphasis on the old Q value and it will take longer time to learn a good policy. But a too high learning rate will put far to much emphasis on new values, making the Q values fluctuate. For static world, a learning rate of 1 will be good (will look the same as the figure from world 1 above). But for a low learning rate, the path will not be as good and the model needs longer training time, see figure below.



9. **Explain how the discount factor γ influences the policy and V-function in each world. Use figures to make your point.**
   The discount factor decides how much the future Q values influence the new Q value. A high discount factor makes the model far sighted, meaning it will take future values into account. A low discount factor does not care so much about future values. In the figure below, a low discount factor was used, as we can see down to the left, it thinks it is just as good to go up and down in a loop as there is to go towards the loop, since it does not care about future values.



10. **Explain how the exploration rate ε influences the policy and V-function in each world. Use figures to make your point. Did you use any strategy for changing ε during training?**
The exploration rate influences how much the model will try new actions. A high exploration rate will make the model try a lot of different action. If the exploration rate is zero, the model will use the optimal actions found so far all the time. We started of exploring much, with an exploration rate of 0.9. After 40% of training we decreased the exploration rate to 0.3. In the figure below, a low exploration rate (0.3) was used for the entire training. As we can see, the robot finds the optimal path, but in the top the policy is not good, since there was no reason to go there we never had the opportunity to learn a good policy there.



11. **What would happen if we instead of reinforcement learning were to use Dijkstra's cheapest path finding algorithm in the "Suddenly Irritating blob" world? What about in the static "Irritating blob" world?**
    In the static world, Dijkstra would find the shortest path without any problems. But if the world is stochastic, and obstacle is not there, it would just take the shortest path to the goal, disregarding the fact that there might be an obstacle there sometimes.

12. **Can you think of any application where reinforcement learning could be of practical use? A hint is to use the Internet.**
    Automated cars (learning to drive), games (alphago), traffic light control, for example.

13. **(Optional) Try your implementation in the other available worlds 5-12. Does it work in all of them, or did you encounter any problems, and in that case how would you solve them?**