

# Intelligent Agents

## Agent Design

Tjalle Galama, 6936385      Ernö Groeneweg, 4279662  
Otto Mättas, 6324363      Shaoya Ren, 6850413  
Vincent de Wit, 6970877

October 17, 2019

Team 1 is designing and implementing a modular agent: the system is decomposed into interacting modules that can be understood on their own. The agent does offline relational representation reasoning on the ontology, and it uses a utility function to make the final decision. The agent can be considered an indefinite horizon planner that looks ahead some finite, but not predetermined, number of stages. Also, knowledge is given to the agent via the ontology.

The agent will take a set of preferences that are put in by the user. These preferences can include a preference (or disliking) for a (or multiple) lecturer(s), topic(s), similarity to another course, methodology, location, time (day and/or slot), as well as whether or not the user would like to take on two or three courses per term. Additionally, the user puts in whatever courses they are already subscribed to, as well as the courses that the user has already finished. These preferences are then turned into queries, which are forwarded to the ontology. For example, if the user wants only courses taught by Lecturer B in time slot C, the query would be "course and isTaughtBy value LECTURER\_B and isTaughtAt value SLOT\_C". The result of this will be a list of courses that satisfy all given conditions. Then, the agent will try to construct as many schedules as possible from this list of courses, taking the previously chosen and already finished courses into account. Then, if there are multiple possible schedules, the agent will feed them into a utility function to determine which of all possible schedules would be the best fit for the user. If there are no schedules possible that fit the user's preferences, the user will be prompted to adapt their preferences accordingly.

Architecturally, this agent is going to function as a hybrid of a goal-based agent and a utility agent. The agent's goal is to construct a schedule based on the user's preferences, however it will always try to construct every possible schedule based on the user's preferences. Then, each course is assigned a single basic utility value, which is increased for every friend of the user who also takes that course, as defined in the ontology. Then, the utility values of each schedule are summed up and the schedule with the highest total utility is recommended. If after applying this utility function there are still multiple possible schedules, one of them is presented to the user at random. The user will then have the option to have another option presented to them, if they so wish.