We present a solution for the real-time simulation of artificial environments containing cognitive and hierarchically organized agents at constant rendering framerates. We introduce a level-of-detail concept to behavioral modeling, where agents populating the world can be both reactive and proactive. The disposable time per rendered frame for behavioral simulation is variable and determines the complexity of the presented behavior. A special scheduling algorithm distributes this time to the agents depending on their level-of-detail such that visible and nearby agents get more time than invisible or distant agents. This allows for smooth transitions between reactive and proactive behavior. The time available per agent influences the proactive behavior, which becomes more sophisticated because it can spend time anticipating future situations. Additionally, we exploit the use of hierarchies within groups of agents that allow for different levels of control. We show that our approach is well-suited for simulating environments with up to several hundred agents with reasonable response times and the behavior adapts to the current viewpoint.

**Key words:** Level of detail – multiagent systems – behavioral modeling
On the behavioral layer, various studies have been done on different types of agents [29] with different architectural approaches [8, 14, 20]. Hierarchical sensors, actions, and contexts that allow more complex behaviors and group engagement were discussed by Atkin et al. [3]. Group behavior has also been thoroughly investigated in [23, 32]. Musse and Thalmann also presented a hierarchical model for simulating virtual human crowds [24]. All of these models rely on the reactive agent concept, whereas Funge introduced a cognitive modeling language, which easily generates sophisticated behavior of individuals through a knowledge representation that also allows for reasoning and planning in addition to reactive behavior [15]. Canamero presented an approach for motivational behavior [9], Aylett et al. presented motivations that drive the behavior and continuous planning in groups [4], and Grosz et al. discussed planning within groups of agents [17]. Bruderlin et al. [7] as well as Isla et al. [20] exploited hierarchies within an agent, while Atkin et al. presented a system that makes use of command hierarchies within groups of agents [3]. O’Hara proposed a system that automatically generates hierarchies of stable subgroups for Reynolds flocking algorithm [26] from which some concepts will be applied to our approach.

With respect to LOD on the cognitive level, O’Sullivan et al. present a framework that allows for LODs within geometry, motion, and even on the cognitive level [27]. Their approach uses role-passing [21] to adapt a character’s possible behavior depending on its LOD. An approach by Musse et al. [22] introduces three different levels of autonomy for an virtual character: guided, programmed, and autonomous. However, these levels are only compared to each other and they do not infer on switching from one level to another automatically.

We combine the approaches of Funge [15], O’Hara [26], and Musse et al. [22] and extend it by introducing level-of-detail with smooth transitions from purely controlled over reactive to proactive behavior within a real-time environment. Additionally, we make use of hierarchies within groups of agents by passing the control from one character to another within the hierarchy.

3 Problem description

The simulation of intelligent characters has become very interesting for the film industry in order to populate sets with artificial characters. Such high-quality simulations can be simulated offline and are therefore not restricted with regard to computation time. In comparison, real-time simulations, such as games, must simulate the whole world within a few milliseconds. In order to provide an acceptable level of intelligence, one needs as much time as possible. These two conflicting requirements demand that a tradeoff be made between time spent for simulation and the quality of “thinking.”

The time per frame is determined by the framerate, which should be at least interactive. This time is split up into rendering, kinematic and physical calculations, and behavioral simulation, as shown in Fig. 1. This article only addresses the latter while using concepts known from real-time rendering. The key is using a level-of-detail approach that distributes the limited, available amount of time such that the quality of the visual impression is as high as possible.

This visual impression depends on the rendering quality as well as on the presented behavior, which should be as intelligent as possible. Intelligence means that some amount of time is invested in “thinking,” which not only generates reactive, but