# 2.1 Problem description

The problem at hand is a minimization problem. The goal is to keep costs to a minimum within a defined supply chain. Within the supply chain, each actor (producer, supplier, etc.) has an agent that can trigger an order at the upstream actor. Each of these agents tries to minimize not only the costs incurred by itself, but the total sum of all costs incurred by the actors in the supply chain. Costs can arise both from the stockpiling of products and from unsatisfied demand. In every level of the supply chain at each time step, four events happen:

1. Previous orders are received (according to the lead-times) from the upstream actor.
2. Order size received from the downstream actor.
3. The received order is fulfilled from on-hand inventory (if possible).
4. The Agent decides about placing order for stock replenishment

The supply chain represents the system, which can assume different states. These states result from the inventory position of the individual actors at a specified time and each state is represented by a vector with these values as elements. Since the individual elements of the vector can theoretically assume arbitrarily large or small values, the values which represent the inventory position need to be transformed into a coding first to generate a limited number of possible states. This results in n over k (binomial coefficient) different possible states, where n represents the number of different coded states and k represents the number of different actors within the supply chain. The initial state for the system is {12, 12, 12, 12}, meaning that each actor has an inventory position of 12 units.

In this problem, the agent has the possibility to react to the environmental state of the supply chain by placing an order with the upstream actor. The ordered quantity is represented by two different variables. The variable X represents the demand of the downstream actor. If, for example, the customer places an order with the retailer for 10 units, the variable X for the retailer takes the value 10. In turn, the variable Y indicates the deviation of the own order at the upstream actor from the order received from the downstream actor. This means that if the value of Y is -1, the agent will order one less unit from the upstream actor than was requested from it by the downstream actor. With a value of +1, the agent orders one unit more, and with a value of 0, it orders exactly as many units as are ordered from it by its downstream actor. The agents can now set the values for Y as an action. These values for Y can theoretically be between minus X and plus infinity. Simulations can be used to find out to which range the values for Y can be fixed without influencing the result in any essential way.

The transition probabilities are based on the Q-values, whereby, since this is a minimization problem, the action that promises the lowest Q-value is always chosen. At the beginning, the probabilities for each action are the same. By estimating the Q-values, new values are gradually learned for the different actions, resulting in different transition probabilities. In this learning phase, the agent selects with a certain probability the action that promises the lowest Q-value (exploitation) and with a certain probability a randomly chosen action (exploration). The initial probabilities are 0.98 for exploration and 0.02 for exploitation. They always add up to 1. The agent therefore is more likely to explore first because his lack of knowledge about the environment. The relationship between exploitation and exploration shifts during the learning process, with the probability of exploitation increasing linearly and the probability of exploration decreasing linearly. After completing the learning process, the agent applies a greedy search and according to the systems state always chooses the action that grants him the lowest Q value.

The reward the agent receives from the environment is calculated using a reward function. In this function, both the inventory position and the backorder are taken into account. These two values are weighted with a cost factor each, alpha and beta, and summed up. It is important here that the costs for all actors involved in the supply chain are calculated and then completely aggregated. This results in an agent wanting to minimize not only the costs incurred within their own operations but also the costs of all parties involved in the supply chain. The two cost factors alpha and beta can be individually chosen here in such a way that they correspond to the costs incurred in the real system.