Knowledge Development in Games of Imperfect Information

Sjoerd Druiven

May 2002

Institute for Knowledge and Agent Technology
University Maastricht

Artificial Intelligence
University of Groningen

supervised by:
H.H.L.M. Donkers M.Sc. & Dr.ir. J.W.H.M. Uiterwijk
Department of Computer Science
Faculty of General Sciences
University Maastricht

Dr. L.C. Verbrugge
Artificial Intelligence
Faculty of Behavioral and Social Sciences
University of Groningen

Acknowledgments

This thesis is the result of a research project which started about 10 months ago. With the contributions of many people, in Groningen and Maastricht, I have been able to come to this result. First of all I thank Rineke Verbrugge and Jeroen Donkers for their supervision of the project. I thank both for their time and confidence. Rineke en Jeroen have given me the freedom to investigate my own ideas about game of imperfect information. Furthermore I thank Jos Uiterwijk for the supervision in Maastricht and for the confidence to have me come over.

I thank Barteld Kooi and Hans van Ditmarsch for their time and enthusiasm. Ariel Rubinstein has given me some of his valuable time, for which I thank him. I thank Fiona Douma for listening to my ideas on card games and the discussion on the game *Kwartet*.

In Maastricht, I did my research at the Institute for Knowledge and Agent Technology. I thank all its members for the warm welcome I had and I especially thank Rens and Michel. In Maastricht, Bob, Gijs, Helen en Karin, made it possible for me to come over and have a pleasant stay in Maastricht. For this I thank them.

Back in Groningen, Artificial Intelligence gave me a place to work. Floris shared his room with me for four months. In Groningen I stayed with my brother Hilbrand. The evening before I gave my presentation in Maastricht, Egon and Cathelijne had me over for dinner and we discussed my work. For all this I thank them.

At last I thank my mother Engelien and Jos for their confidence and freedom. I especially thank Ali and my father Henk for their support and confidence, even in times when they deserved all the attention. I thank Iris for her sincere interest in my research, her understanding and support. She has helped through the last two months of the writing my thesis.

Contents

Introduction 1

1 Games & Knowledge 3
1.1 Introduction 3
1.2 Game Theory 3
1.2.1 Game of Imperfect Information 3
1.2.2 Strategies 5
1.2.3 Example Matching Pennies 7
1.2.4 Rationality 9
1.3 Analysis of a Card Game 11
1.3.1 Game & Strategies 12
1.3.2 Rationality 15
1.3.3 Knowledge 18
1.3.4 Conclusions of a Card Game 20
1.4 Conclusions 21

2 Modeling Knowledge 23
2.1 Introduction 23
2.2 Dynamic Epistemic Logic 23
2.2.1 Epistemic Logic 24
2.2.2 Dynamic Epistemic Logic 25
2.2.3 Example Muddy Children Puzzle 27
2.2.4 Baltag 30
2.2.5 Conclusions Dynamic Epistemic Logic 31
2.3 Analysis of a Card Game 31
2.3.1 Game & Kripke Model 31
2.3.2 Definite Knowledge 32
2.3.3 Strategic Knowledge 35
2.3.4 Conclusions of a Card Game 39
2.4 Conclusions 39

3 Dynamic Epistemic Games 41
3.1 Introduction 41
3.2 Dynamic Epistemic Games 42
3.2.1 Dynamic Epistemic Logic 42
3.2.2 Dynamic Epistemic Game 44
3.2.3 Strategies 47
3.2.4 Example Hexas 47
3.2.5 Comparison with Game Theory 52

3.2.6 Conclusions Dynamic Epistemic Games 54
3.3 Knowledge 55
3.3.1 Game Knowledge 55
3.3.2 Definite Knowledge 55
3.3.3 Strategic Knowledge 56
3.4 Analysis of the Card Game 58
3.4.1 Game & Strategies 60
3.4.2 Strategic Knowledge 63
3.4.3 Conclusions of the Card Game 66
3.5 Discussion 66
3.6 Further Research 68
3.7 Conclusions 69

Conclusion **70**

Bibliography **73**

Introduction

A card game is played. The game consists of a deck of playing cards, some players and a table. Three cards are taken from the deck and are laid on the table, facing down. The other cards are dealt over the players. The goal of the game is to learn which cards are lying on the table. After the dealing of the card a player asks another player "Are you holding the four of spades or the eight of clubs ?". The second player answers the question truthfully. Suppose the players have some knowledge about the questioning behavior of player 1. What do the players know after the question and answer ? Such questions, and the answers to it belong to the main theme of this master thesis.

The game played is a game of imperfect information. In games of imperfect information, players are uncertain about the precise state of the game. The players do not know the state of the game, because the players are imperfectly informed about the events that happened in the game. For instance, in the card game, the players do not know how the cards have been dealt over the players. Therefore the term *game of imperfect information.*

There exists a clear distinction between games of imperfect information and games of incomplete information. A game is of incomplete information, the player are not completely informed about the game. For instance, the players are not informed about the winning conditions of the game. Or the players do not know what the available actions are. The knowledge the players have about the game, what we call game knowledge, is incomplete. Games of incomplete information are not studied in this thesis.

In games of imperfect information, we attribute distinct forms of knowledge to the players. In this thesis we introduce a distinction of the forms of knowledge players have. We distinguish three forms of knowledge: game knowledge, definite knowledge and strategic knowledge. Game knowledge is the knowledge the players have about the game they are playing. In the card game the players know which cards belong to the deck and how the cards are dealt over the players and the table. Definite knowledge is the knowledge players develop as a consequence of explicit information exchange. In the card game the players learn about the deal of cards through the question and answer. The question and answer explicitly inform the players and the players develop definite knowledge.

Suppose every player knows that the player to ask the question only asks about he four of spades if she is holding the ace of hearts. When the player asks her question, every player learns that she must be holding the ace of hearts. We call this form of knowledge strategic knowledge. The players learn about the deal of cards, because they have knowledge about the used strategy. The goal of this master thesis is to model the definite and strategic knowledge players develop during games of imperfect information.

In the first chapter, we give an introduction to game theory. Given this theory, game, definite and strategic knowledge are precisely described. Definite knowledge in games of imperfect information is elegantly modeled by dynamic epistemic logic, as shown by Hans van Ditmarsch in *Knowledge Games* [20]. Strategic knowledge has not been given much attention by the scientific community as definite knowledge. In chapter 2, we introduce a dynamic epistemic logic. In the dynamic epistemic logic, the definite knowledge players develop during a game of imperfect information can be modeled. We conclude that strategic knowledge cannot be modeled by dynamic epistemic logic to satisfaction. In chapter 3 we present our main contribution, namely dynamic epistemic games. By modeling games of imperfect information as dynamic epistemic games, strategic knowledge can be modeled to satisfaction.

Chapter 1

Games & Knowledge

1.1 Introduction

A game is a strategic interaction. Game theory can be used to model and analyze this interaction. We are concerned with a specific type of game, a game of imperfect information. With respect to game theory, we have several interests in these games.

First of all, we are interested in what precisely the knowledge of the players is about the game they are playing. We want to know what the game knowledge of the players is. Secondly, we are interested how players exchange information in a game of imperfect information and how this alters their knowledge. Thirdly, we are concerned with the strategies of the players, and how knowledge about the used strategies can influence the knowledge of the players about the state of the game. In other words, how does strategic knowledge develop ?

In this chapter we want to clarify what strategies, definite and strategic knowledge are in a game of imperfect information. We have worked out several examples of strategic knowledge in simple card game. In the examples the concept of strategic knowledge is worked out in detail. First, we give an introduction to game theory, games of imperfect information and game-theoretic concepts to be able to give a precise insight to game, definite and strategic knowledge later on in the chapter.

1.2 Game Theory

In this section we give an introduction to games of imperfect information and related concepts, such as as a pure strategy, a mixed strategy, a strategy profile and an outcome of a game given a strategy profile. All definitions are after the example of *A Course in Game Theory* by Martin J. Osborne and Ariel Rubinstein [17].

1.2.1 Game of Imperfect Information

In this subsection we introduce the model of a game of imperfect information. First we want to clarify the distinction between a game and its game-theoretic model.

Game of Imperfect Information An informally described game can be formally modeled by a game model. First we need to clear up the uses of the term *game of imperfect information* to avoid confusion. The term *game of imperfect information* has two meanings. First of all, a game of imperfect information is the game actually played. The game may consists of cards, money, a playing board or other attributes and is played by a certain number of players. Secondly, a game of imperfect information is also the term used for the *model* of the game. The model of the game consist of a set of players, a set of histories and some functions. Given the clarification we introduce the model of a game of imperfect information.

Definition 1 (Game of Imperfect Information)
A *game of imperfect information* is a tuple $(N, H, P, f_c, (\mathcal{I}_n)_{n \in N}, u)$.

- N is a finite set of *players*;
- H is a finite set of *histories*. A *history* is a sequence of actions $(a^k)_{k=1 \ldots K}$. For the set of histories H the following properties hold:
  - The empty sequence $\varnothing$ is a member of H;
  - Every subhistory of a history in H is also a member of H, i.e. if $(a^k)_{k=1 \ldots K} \in H$ and $L < K$ then $(a^k)_{k=1 \ldots L} \in H$.

  Furthermore we define:

  - A history $(a^k)_{k=1 \ldots K} \in H$ is *terminal* iff there is no a^{K+1} such that $(a^k)_{k=1 \ldots K+1} \in H$ holds. The set of terminal histories is denoted by Z;
  - An action a^{K+1} is *available* after a history $(a^k)_{k=1 \ldots K} \in H$, if the history $(a^k)_{k=1 \ldots K+1}$ is a member of H. The set of available actions after history h is $A(h)$.
- The *player function* $P : N \cup \{c\} \to 2^{H/Z}$ is a function which assigns to each player and *chance*, denoted by c, a set of non-terminal histories. $P(c) \subseteq H$ is the set of histories in which *chance* decides which action is taken. $P(n) \subseteq H$ is the set of histories in which player n decides which action is taken. P partitions the set of non-terminal histories, i.e. there exists no history h such that $h \in P(n)$, $h \in P(m)$, and $n \neq m$.
- To each history h such that $h \in P(c)$, f_c assigns a probability distribution $f_c(\cdot|h)$ over set of the available actions $A(h)$. $f_c(a|h)$ is the probability that action a occurs after history h;
- $\mathcal{I}_n$ denotes the information partition of player n. The information partition $\mathcal{I}_n$ of a player divides the set of histories in which the player has to take an action, $P(n)$, into several information sets I_n. For the histories h, h' of an information set I_n it holds that available actions are the same, i.e $\forall h' \in I_n : A(h) = A(h')$.
- The *payoff function* u is a function which assigns for each player n at each terminal history $h \in Z$ a payoff $u(n, h) \in \mathbb{R}$.

We distinguish two types of actions, actions taken by the players and actions determined by *chance*. *Chance* is used to model for instance the rolling of a dice or the dealing of cards. When modeling a game, first all actions in the game are determined. A sequence of actions is a history. A history determines the state of the game. Therefore we can view histories as game state and vice versa. After every sequence the player to take an action is determined. This can be one of the players, or *chance*.

On the set of histories, the information sets of the players are determined. Histories are in the same information set if the player to take an action cannot distinguish the histories. A player cannot distinguish sequences of actions if the player is imperfectly informed about which actions were taken.

Game Tree Games of imperfect information can be viewed as a tree. A node in a tree is a state of the game. Every branch corresponds to an action. The path to the node is the history, the sequence of actions which lead to that state of the game. Each node is labeled with the player to take an action. If two nodes are in the same information set, the nodes are connected by a dashed-line. The leaves of the tree correspond to the terminal histories. The leaves are labeled with the payoffs respectively giving to every player. When modeling games of imperfect information we graphicly display the model of the game by a game tree. We refer to the game tree as the model of the game.

Game Knowledge Game knowledge is knowledge players have about the game they are playing. Here we present an overview of the game knowledge of the players of a game of imperfect information. If a player's game knowledge is incomplete, the game is of incomplete information. The players commonly know the set of players, the set of histories, the players function, the information partition, the probability distribution f_c and the payoff function. In other words, the players commonly know the game they are playing. Furthermore, it is common knowledge that the players have this game knowledge.

The players do not necessarily know the state of the game. This has some consequences with respect to the game knowledge of the players. The players do not necessarily know the player to take an action. A player does also not necessarily know the actions available to the player to take an action. The player to take an action does know the actions available to him.

Two remarks: First, by assuming the players have this knowledge, we implicitly assume the players are perfect logicians. Secondly, if the players do not commonly know the game they are playing, the game is of incomplete information.

1.2.2 Strategies

A strategy is recipe for a player to play a game. Here we introduce a pure strategy and a mixed strategy in a game of imperfect information. Mixed strategies involve probabilities, in contrast with pure strategies.

Pure Strategy A pure strategy of a player in a game of imperfect information is a prescript that specifies the action chosen after every history in which the player has to take an action. A strategy of a player prescribes to take the same action after histories belonging to the same information set, as the player cannot distinguish the histories.

Definition 2 (Pure Strategy)
Given a game of imperfect information $(N, H, P, f_c, (\mathcal{I}_n)_{n\in N}, u)$, a *pure strategy* of a player n is a function $s_n : P(n) \to \{a|h \in P(n) \text{ and } a \in A(h)\}$, such that two histories are assigned the same action if the histories are in the same information set, i.e. if $h, h' \in I_n$ and $I_n \in \mathcal{I}_n$ then $s_n(h) = s_n(h')$. s_n assigns to each history h of player n an available action $a \in A(h)$. The set of pure strategies of player n is denoted by S_n.

A *pure strategy profile* $\boldsymbol{s}$ is a list $(s_1, \ldots, s_{|N|})$ consisting of a pure strategy for each player. $\boldsymbol{s}_{-n}$ is the list consisting of a pure strategy for each player except for player n. $\boldsymbol{s}_n$ is the strategy of player n according to the strategy profile. $(\boldsymbol{s}_{-n}, \boldsymbol{s}_n)$ is equal to the pure strategy profile $\boldsymbol{s}$. The set of all pure strategy profiles is denoted $\boldsymbol{S}$.

A pure strategy beforehand determines for a player which actions to take for the whole game. Given a pure strategy profile determines, the outcome of the game can be calculated, as the profile determines which actions will be taken in every history.

Mixed Strategy If a player uses a mixed strategy, she randomizes over her pure strategies according to some probability distribution. The randomization is done before the game is played, and the player plays the game according to the resulting pure strategy. The other players are kept ignorant about the used pure strategy.

Definition 3 (Mixed Strategy)
Let a game of imperfect information $(N, H, P, f_c, (\mathcal{I}_n)_{n\in N}, u)$ be given. A *mixed strategy* σ_n of a player n is a probability distribution over the set of pure strategies S_n of player n. $\sigma_n(s_n)$ is the probability of pure strategy s_n, given the mixed strategy σ_n.

A *strategy profile* $\boldsymbol{\sigma}$ is a list $(\sigma_1, \ldots, \sigma_{|N|})$ consisting of a strategy, pure or mixed, for each player. A strategy profile has the same properties as a pure strategy profile. $\boldsymbol{\sigma}_{-n}$ is the list consisting of a strategy for each player except for player n. $\boldsymbol{\sigma}_n$ is the strategy of player n according to the strategy profile. $(\boldsymbol{\sigma}_{-n}, \boldsymbol{\sigma}_n)$ is equal to the strategy profile $\boldsymbol{\sigma}$.

Outcome Given a pure strategy profile, the outcome of the game of imperfect information can be determined. The outcome is a probability distribution over the set of terminal histories. The probability assigned to a terminal history is the probability that the history will be reached if all players use strategies according to the strategy profile $\boldsymbol{\sigma}$. The expected payoff is the expected value of the payoff function, given this probability distribution.

Definition 4 (Outcome of a pure strategy profile)
Let a game of imperfect information $(N, H, P, f_c, (\mathcal{I}_n)_{n\in N}, u)$ and a pure strategy profile $\boldsymbol{s}$ be given. The *outcome* $O(\cdot|\boldsymbol{s})$ of the game is a probability distribution over the set of terminal histories Z, such that every terminal history z is assigned the probability that history z will be reached if all players use the pure

strategies according to the pure strategy profile $\boldsymbol{s}$. $O(z|\boldsymbol{s})$ is the probability that terminal history z will be reached if the players play according to strategy profile $\boldsymbol{s}$.

Let a terminal history $(a^k)_{k=1\ldots K}$ be *consistent* with strategy profile $\boldsymbol{s}$ if for every $k : 0 < k \leq K, (a^k)_{k=1\ldots K} \in P(n)$ and $n \neq c$ holds $\boldsymbol{s}_n(a^1 \ldots a^k) = a^{k+1}$. The outcome of the game given strategy profile $\boldsymbol{s}$ is now defined such that inconsistent terminal histories are assigned a probability of zero and a consistent terminal history $(a^k)_{k=1\ldots K}$ is assigned a probability equal to $\prod_{\{0<k\leq K|(a^1\ldots a^{k-1})\in P(c)\}} f_c(a^k|(a^1 \ldots a^{k-1}))$.

The *expected payoff* $E_u(n, \boldsymbol{s})$ of player n given strategy profile $\boldsymbol{s}$ is $\Sigma_{h\in Z} u(n,h) \cdot O(h|\boldsymbol{s})$.

Using the outcome and expected payoff in pure strategies, the general outcome and expected payoff of a game of imperfect information is determined.

Definition 5 (Outcome)
Let a game of imperfect information $(N, H, P, f_c, (\mathcal{I}_n)_{n\in N}, u)$ and a strategy profile $\boldsymbol{\sigma}$ be given. The *outcome* $O(\cdot|\boldsymbol{\sigma})$ of the game is a probability distribution over the set of terminal histories Z.

The probability of a terminal history z given a strategy profile $\boldsymbol{\sigma}$ is a weighted sum over the set of pure strategy profiles $\boldsymbol{S}$ of the probabilities of the terminal history z given each pure strategy profile $(s_1, \ldots, s_{|N|})$. The probability of terminal history z given the pure strategy profile $(s_1, \ldots, s_{|N|})$ is $O(z|(s_1, \ldots, s_{|N|}))$. The weight of the sum is the probability of the pure strategy profile $(s_1, \ldots, s_{|N|})$ given strategy profile $\boldsymbol{\sigma}$. The probability of the pure strategy profile $(s_1, \ldots, s_{|N|})$ given the strategy profile $\boldsymbol{\sigma}$ is the product of the probabilities of the pure strategies of the pure strategy profile $(s_1, \ldots, s_{|N|})$ given the strategy profile $\boldsymbol{\sigma}$. Let $f_s(s_n|\sigma)$ be the probability of pure strategy s_n given strategy profile $\boldsymbol{\sigma}$. $f_s(s_n|\sigma)$ is 1 if $\sigma_n = s_n$, otherwise $f_s(s_n|\sigma) = \boldsymbol{\sigma}_n(s_n)$. The probability of terminal history z given the mixed strategy profile is $O(z|\boldsymbol{\sigma}) = \Sigma_{(s_1,\ldots,s_{|N|})\in \boldsymbol{S}}(\prod_{n\in N} f_s(s_n|\boldsymbol{\sigma}) \cdot O(z|\boldsymbol{s}))$.

The *expected payoff* $E_u(n, \boldsymbol{\sigma})$ of player n is $\Sigma_{h\in Z} u(n,h) \cdot O(h|\boldsymbol{\sigma})$.

Given a strategy for every player, the expected payoffs can be calculated. The expected payoff can be used to determine whether a strategy of a player is a good strategy or not.

1.2.3 Example Matching Pennies

An example of a game of imperfect information is the game Matching Pennies. The game consists of two players and every player has a penny. Both players choose a side of their coin, by placing the coin on the table and letting the chosen side face up. The choice of the side is made secretly. After both players have chosen a side, their choice is revealed to each other by showing the coins. If the choices are identical, player 2 loses his penny and player 1 wins and ins the two valuable pennies, otherwise the payoffs are vice versa.

Game model The game Matching Pennies can be modeled as the game of imperfect information $(N, H, P, f_c, (\mathcal{I}_n)_{n\in N}, u)$:

- $N = \{1, 2\}$;

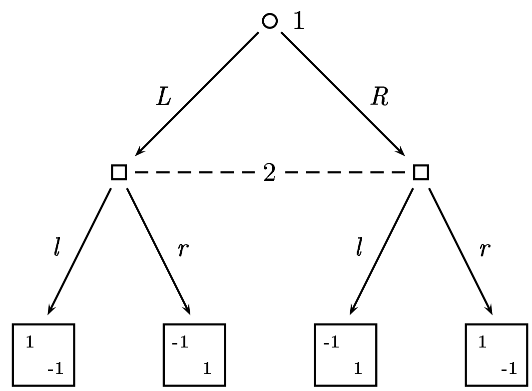

Figure 1.1: Matching Pennies

- $H = \{\varnothing, (L), (R), (L,l), (L,r), (R,l), (R,r)\}$;
- $P(1) = \{\varnothing\}, P(2) = \{(L), (R)\}$;
- $f_c = \emptyset$, no action taken by*chance* exist.
- $\mathcal{I}_1 = \{\{\varnothing\}\}, \mathcal{I}_2 = \{\{(L), (R)\}\}$;
- $u((L,l),1) = 1$, $u((L,l),2) = -1$, $u((L,r),1) = -1$, $u((L,r),2) = 1$, $u((R,l),1) = -1$, $u((R,l),2) = 1$, $u((R,r),1) = 1$, $u((R,r),2) = -1$.

Figure 1.1 represents the game Matching Pennies. In the game Matching Pennies, the players make their move simultaneous. The moves made cannot be modeled as simultaneous actions. The actions are sequential. First player 1 chooses a side of her coin, followed by player 2. Player is not informed about the choice of player 1, therefore the information set of player 2. The leaves of the trees are labeled with the payoffs of the players. The top-left number is the payoff to player 1, the bottom-right is the payoff to player 2.

Strategies As an example we have worked out the pure strategies of the players, the outcome of a pure strategy profile, the expected payoff of the pure strategy profile, a mixed strategy for every player, the outcome and an expected payoff of a strategy profile.. Player 1 has two pure strategies, s_1 and s_1':

$$s_1(\varnothing) = L$$
$$s_1'(\varnothing) = R$$

The pure strategies of player 2 are s_2 and s_2':

$$s_2(L) = s_2(R) = l$$
$$s_2'(L) = s_2'(R) = r$$

The outcome O of strategy profile (s_1, s_2) is:

$$\begin{aligned} O((L,l)|(s_1,s_2)) &= 1 \\ O((L,r)|(s_1,s_2)) &= 0 \\ O((R,l)|(s_1,s_2)) &= 0 \\ O((R,r)|(s_1,s_2)) &= 0 \end{aligned}$$

The expected payoffs of the players given the strategy profile (s_1, s_2) are:

$$\begin{aligned} E_u(1,(s_1,s_2)) &= 1 \\ E_u(2,(s_1,s_2)) &= -1 \end{aligned}$$

A mixed strategy of a player is a probability distribution over her pure strategies. Let the mixed strategy σ_1 of player 1 be such that:

$$\begin{aligned} \sigma_1(s_1) &= \tfrac{3}{4} \\ \sigma_1(s_1') &= \tfrac{1}{4} \end{aligned}$$

Let the mixed strategy σ_2 of player 2 be:

$$\sigma_2(s_2) = \sigma_2(s_2') = \tfrac{1}{2}$$

The outcome O of the game given the strategy profile (σ_1, σ_2) is:

$$\begin{aligned} O((L,l)|(\sigma_1,\sigma_2)) &= \tfrac{3}{8} \\ O((L,r)|(\sigma_1,\sigma_2)) &= \tfrac{3}{8} \\ O((R,l)|(\sigma_1,\sigma_2)) &= \tfrac{1}{8} \\ O((R,r)|(\sigma_1,\sigma_2)) &= \tfrac{1}{8} \end{aligned}$$

The expected payoff of the players given the strategy profile (σ_1, σ_2) are:

$$\begin{aligned} E_u(1,(\sigma_1,\sigma_2)) &= 0 \\ E_u(2,(\sigma_1,\sigma_2)) &= 0 \end{aligned}$$

1.2.4 Rationality

Rational players maximize their own expected payoff. This is the bottom line of rationality. In games, the players often have to make decisions under uncertainty. Osborne and Rubinstein [17] distinguish four forms of uncertainty. The players may be:

- uncertain about the objective parameters of the environment;
- imperfectly informed about events that happen in the game;
- uncertain about actions of the other players that are not deterministic;
- uncertain about the reasoning of the other players.

If a player is uncertain about an aspect of the game, a rational player is assumed to behave as if he has in mind an expectation about the uncertainty.

Game theory tries to determine what rational behavior is in games, with and without uncertainty. Two concepts which are directly related to rational behavior are the concepts of a Nash-equilibrium and strict domination.

Nash-Equilibrium Given a strategy profile, the best-response of a player is the strategy which maximizes her expected payoff of the strategy profile consisting of the ordinal strategy profile substituted with the best-response strategy.

Definition 6 (Best-response)
Let a game of imperfect information $(N, H, P, f_c, (\mathcal{I}_n)_{n \in N}, u)$ and a strategy profile $\boldsymbol{\sigma}$ be given. A *best response* of player n with respect to player m to strategy profile $\boldsymbol{\sigma}$ is a strategy σ_n^* such that for all strategies σ_n of player n holds $E_u(m, (\boldsymbol{\sigma}_{-n}, \sigma_n^*)) \geq E_u(m, (\boldsymbol{\sigma}_{-n}, \sigma_n))$.

The above definition differs from the definition in *A Course in Game Theory* by Martin J. Osborne and Ariel Rubinstein [17]. The difference does not affect the concept of a Nash-equilibrium. If a rational player knows which strategies are used by the other players, the player will use a strategy which is a best-response with respect to herself to the strategies of the other players.

A Nash-equilibrium is a strategy profile, such that all strategies of the strategy profile are best-responses to the strategy profile itself.

Definition 7 (Nash-equilibrium)
Let a game of imperfect information $(N, H, P, f_c, (\mathcal{I}_n)_{n \in N}, u)$ be given. A strategy profile $\boldsymbol{\sigma}^* = (\sigma_1, \ldots, \sigma_{|N|})^*$ is a *Nash-equilibrium* iff for every player n it holds that strategy σ_n is a best response of player n with respect to player n to strategy profile $\boldsymbol{\sigma}^*$.

A Nash-equilibrium is a strategy profile, such that if the players expect the game to be played according the Nash-equilibrium, rational players will stick to their strategy of the Nash-equilibrium. Osborne and Rubinstein give a good comment on the concept of a Nash-equilibrium:

> This notion (i.e. Nash-equilibrium) capture a steady state of the play of a (..) game in which each player holds the correct expectation about the other players' behavior and acts rationally. It does not attempt to examine the process by which a steady state is reached.

In other words, if the players expect from each other a game will be played according to a Nash-equilibrium, it is rational to play the game according to the Nash-equilibrium. How these expectations are formed is another question, which we will not go into.

Strict Domination Another concept directly related to rational behavior is strict domination. First we define a guaranteed payoff, before we define strict domination. Given a strategy of a player, the guaranteed payoff is the worst possible payoff to the player.

Definition 8 (Guaranteed Payoff)
Let a game of imperfect information $(N, H, P, f_c, (\mathcal{I}_n)_{n \in N}, u)$ and the mixed strategy σ_n be given. A payoff u is *guaranteed* by σ_n if there exists no strategy profile $\boldsymbol{\sigma}$ such that $\boldsymbol{\sigma}_n = \sigma_n$ and the expected payoff of the strategy profile $\boldsymbol{\sigma}$ is lower, i.e. $E_u(n, \boldsymbol{\sigma}) < u$.

A pure strategy is strictly dominated if there exists a mixed strategy which guarantees a higher payoff.

Definition 9 (Strict Domination)
Let a game of imperfect information $(N, H, P, f_c, (\mathcal{I}_n)_{n \in N}, u)$ be given. A pure strategy s_n of player n is *strictly dominated* iff there exists a strategy σ_n (mixed or pure) of player n which guarantees a higher payoff. I.e. for every strategy profile $\boldsymbol{\sigma}$ (mixed or pure) such that $\boldsymbol{\sigma}_n = s_n$ it holds that $E_u(n, \boldsymbol{\sigma}) \leq E_u(n, (\boldsymbol{\sigma}_{-n}, \sigma_n))$.

A rational player will never use a strictly dominated strategy. Furthermore, when using mixed strategies, rational players always assign zero probabilities to strictly dominated strategies.

Example Matching Pennies Let the mixed strategy σ_1^* of player 1 be:

$$\begin{aligned} \sigma_1^*(s_1) &= \tfrac{1}{2} \\ \sigma_1^*(s_1') &= \tfrac{1}{2} \end{aligned}$$

Let the mixed strategy $\sigma_2 j$ of player 2 be:

$$\begin{aligned} \sigma_2^*(s_2) &= \tfrac{1}{2} \\ \sigma_2^*(s_2') &= \tfrac{1}{2} \end{aligned}$$

The outcome of the game given the strategy profile $(\sigma_1 j, \sigma_2 j)$ is:

$$\begin{aligned} O((L,l)|(\sigma_1^*, \sigma_2^*)) = O((L,r)|(\sigma_1^*, \sigma_2^*)) &= \tfrac{1}{2} \\ O((R,l)|(\sigma_1^*, \sigma_2^*)) = O((R,r)|(\sigma_1^*, \sigma_2^*)) &= \tfrac{1}{2} \end{aligned}$$

The expected payoff to the players is:

$$\begin{aligned} E_u(1, (\sigma_1^*, \sigma_2^*)) &= 0 \\ E_u(2, (\sigma_1^*, \sigma_2^*)) &= 0 \end{aligned}$$

Suppose the game Matching Pennies is played according to the strategy profile $(\sigma_1 *, \sigma_2 *)$. If a player deviates from her strategy, the player will never increase her expected payoff. As a matter of fact, the expected payoff of player 2 is always zero, if player 1 uses mixed strategy σ_1^*. The same holds for player 1 if player 2 uses strategy σ_2^*. As the players cannot increase their payoff, by deviating from their strategies, every strategy of a player is a best-response to the strategy-profile. Therefore the strategy profile is a Nash-equilibrium. In general it holds that if a Nash-equilibrium is found in mixed strategies, the mixed strategies used by the players are such that an opponent becomes indifferent about her used strategy. Every strategy of the player will result in the same expected payoff.

In the game Matching Pennies, no pure strategy is strictly dominated. Every pure strategy can be the rational way to behave in some situations. If a pure strategy would never have an expected payoff higher than zero, this strategy would be strictly dominated, as players 1 and 2 can guarantee a payoff of zero by respectively using σ_1^* and σ_2^*.

1.3 Analysis of a Card Game

In this section we will give an analysis of a card game, called Q-A. The analysis will go into available strategies to the players, rational behavior and knowledge development in the game Q-A. Q-A is a card game in the spirit of hexa. The card game is a very similar to the card game hexa, introduced by Van Ditmarsch in Knowledge Games [20] and a card game described by Ariel Rubinstein [18]. The game consists of three cards, two players and a table. The cards are dealt over the players and the table. The goal of the game is to correctly guess the card on the table.

1.3.1 Game & Strategies

In this subsection we present the description of the game. We model the game and go into the available strategies to the players. Player 1 and 2 will be addressed with she and he respectively.

Description of Q-A The game consists of two players, a table, three cards and a pot. To play the game, players pay the pot an ante of $ 4. The cards are a red, a blue and a white card. Both players are dealt a card and the third card is put on the table, facing down. The players can only see their own card.

After the dealing of the cards, player 1 asks player 2 a question. Player 1 has a choice of three possible questions: 1. "Are you holding the red card?"; 2. "Are you holding the white card?" and 3. "Are you holding the blue card?". Player 1 is the only player to ask a question. Then, player 2 answers player 1's question. Player 2 must answer the question of player 1 truthfully.

When player 2 has answered player 1's question, both players make a guess about the card on the table and the card on the table is shown to the players. The payoff to the players depends on the guess of the players. If both players guess correctly, the pot is split up by the players and both receive their ante back of $ 4. If only one player guesses correctly, she receives the whole pot of $ 8. The incorrectly guessing player is punished. As a punishment, the player has to burn $ 20. If both players guess incorrectly, both players burn $ 20, together with the pot of $ 8.

We call the game Q-A: Question-and-Answer.

Game Model of Q-A We model the game Q-A as an game of imperfect information to determine the pure strategies available to the players. The corresponding game-tree is described step-by-step, beginning at the dealing of the cards and finishing with the payoffs to the players. Figure 1.2 displays the game-tree of the game Q-A. We have left out branches for reasons of readability.

Dealing of the cards A deal of cards is denoted by a sequence of letters; the sequence rwb denotes that player 1 is holding the red card, player 2 is holding the white card and the blue card is lying on the table. The first character denotes the card player 1 is holding, the second character the card player 2 is holding and the last character denotes which card is lying on the table. The number of possible deals of cards is $3! = 6$.

Every deal of cards is equally likely to be dealt. Players can only see their own card. A player cannot distinguish deals of cards such that the player is dealt the same card. Player 1 cannot distinguish rwb from rbw, wrb from wbr and brw from bwr Player 2 cannot distinguish rwb from bwr, wrb from brw and rbw from wbr

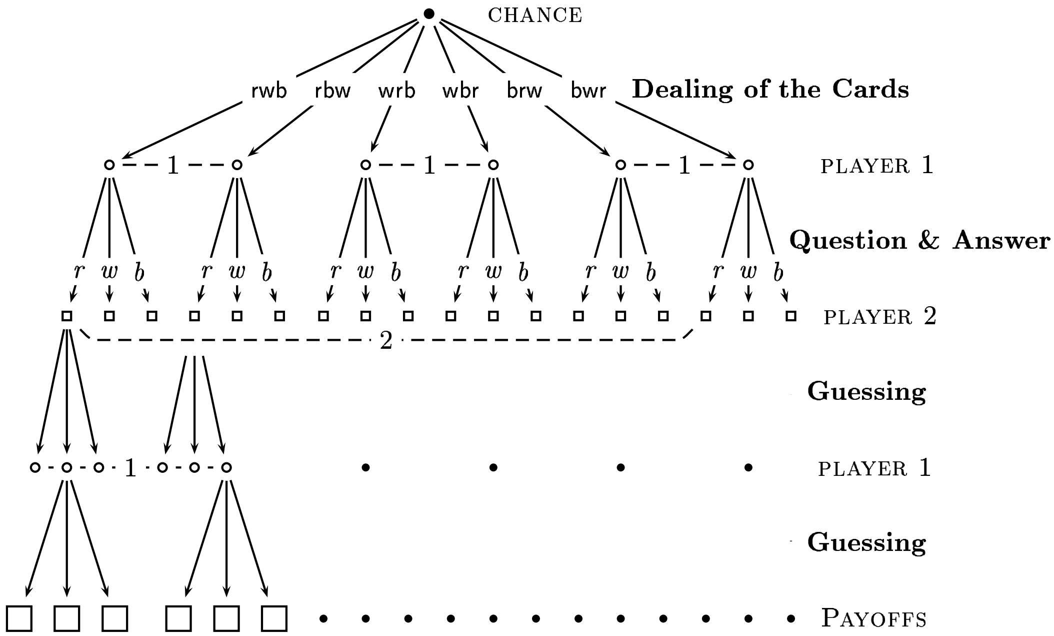

Figure 1.2: Q-A

The first layer of the game tree models the dealing of the cards. The dealing of the cards is an action taken by *chance*. Although not displayed in the game-tree, every deal of cards is assigned a probability of $\frac{1}{6}$.

After the deal of cards player 1 has to take an action. The information sets of player 1 are represented by the dashed lines connecting the game states labeled with 1. Two game states are in the same information set of player 1 if player 1 is holding the same card in both game states.

Question & Answer The questions are respectively denoted by r, w and b. Player 1 is allowed to asks about her own card. For instance she is allowed to ask about the red card, when holding the red card. The answer to the question of player 1 is determined by the deal of cards. The second layer of the game tree models the question and answering. The answer of player 2 is not modeled as a separate action in the game tree, because the answer is determined by the deal of cards. If the answer of player 2 would be modeled as an action taken by player 2, player 2 would be able to choose which answer to give. Player 2 cannot choose over his answers, as he is obliged to answer truthfully.

The questions and answers are announced publicly, it is common knowledge between the players what the question and answer are. After the question and answer, players cannot distinguish game states such that the player is holding the same card, the same question was asked and the same answer was given. For instance, after player 1 has asked about the red card player 2 cannot distinguish the game state such that the deal of cards is rwb and the game state such that the deal of cards is bwr.

After the question and answer it is player 2's turn. The game states that player 2 cannot distinguish are in the same information set of player 2. As an example we have worked out one information set. The information set is represented by the dashed line connecting two game states, labeled with player 2. The two game states of the information sets are the states described in the previous paragraph.

Guessing The third and fourth layer model the guessing about the card on the table. In the game, the guesses are made simultaneous. The simultaneous moves are modeled as sequential actions in the game tree. So in the actual game, player 1 and 2 make their guess at the same time. In the game tree on the other hand, first player 2 makes his guess, followed by player 1. Players are allowed to deliberately guess incorrectly by guessing the card on the table is the card the player is holding.

As an example, we have worked out two branches in the game tree. In these branches player 1 is holding the red card and asked about the red card.

Payoffs The payoffs correspond to the amount of profit the players have made at the end of the game. For instance, the deal of cards is rwb and player 1 asked about the red card. Furthermore, player 1 correctly guessed the blue card was lying on the table, while player 2 incorrectly guessed the red card was lying on the table. The payoffs are as follows.

Both players payed an ante of $ 4 to the pot before the game. At the end of the game, player 1 receives the whole pot of $ 8 and makes a profit of $ 4. Her payoff is 4. Player 2's guess was incorrect, and therefore he is punished. Player 2 looses $ 20. Player 2's loss is his ante of $ 4 plus the $ 20 is $24. Player 2's payoff is -24. If both players would have guessed correctly, the payoffs for both player would have been 0. If both players guess incorrectly, the payoffs of the players is -24.

Strategies A strategy of a player is a choice of action in each information set of the player. In total, player 1 has 18 information sets. Player 1 has 3 information sets in which she asks a question. Every information set corresponds to a card player 1 can hold. After the guess of player 2, player 1 has 15 information sets. The 15 information sets correspond to distinct combination of the card player 1 is holding, the question of player 1 and the answer of player 2. In every information set of player 1, 3 actions are available. Thus the number of distinct pure strategies of player 1 is 3^{18}, the number of available actions times the number of information sets.

A pure strategy of player 1 is a recipe prescribing the question to ask given the card player 1 is holding and the guess given the card player 1 is holding, the question asked and the answer of player 2.

Player 2 has 9 information sets. Every information set of player 2 corresponds to a distinct combination of the card player 2 is holding and the question asked by player 1. In every information set player 2 can make three guesses, and so player 2 has 3^9 distinct pure strategies. A pure strategy of player 2 prescribes player 2 which guess to make given the card player 2 is holding and question of player 1.

The number of pure strategy profiles is the product of the number of pure strategies for each player. The number of distinct pure strategy profiles $3^{18} \times 3^9 = 387.420.489 \times 19.683 = 3^{27} \approx 8 \cdot 10^{12}$.

1.3.2 Rationality

In this subsection we will analyze what rational behavior is in the card game. What are strictly dominated strategies and what are the Nash-equilibria of the game? We will show that it is strictly dominated for player 1 to ask player 2 about the card player 1 is holding herself. Furthermore we give an example of a Nash-equilibrium for the game Q-A.

Asking your own card We want to show that it is strictly dominated for player 1 to ask about her own card. Let an aoc-strategy be a pure strategy of player 1 such that given at least one of the three cards, player 1 asks about her own card. For instance, always asking about the red card is an aoc-strategy. Given the red card, player 1 asks about her own card. We show that it is strictly dominated for player 1 to ask ask about her own card by showing that all aoc-strategies are strictly dominated. By showing that every aoc-strategy is strictly dominated, we exclude every possible strategy such that player 1 asks about her own card.

Strict Domination Let the pure strategy s_n be given. Furthermore, let the pure strategy profile $\boldsymbol{s}$ be such that for every player m it holds that $\boldsymbol{s}_m$ is a best-response of player m with respect to player n to the strategy profile $\boldsymbol{s}$. The pure strategy s_n is strictly dominated if the expected payoff $E_u(n, \boldsymbol{s})$ of the strategy profile $\boldsymbol{s}$ is lower than a payoff guaranteed by a mixed strategy σ_n.

We will show some pure strategies are strictly dominated. Given the above alternative definition of strict domination we can show a pure strategy s_n is strictly dominated by using the following recipe:

1. For the pure strategy s_n construct the pure strategy profile $\boldsymbol{s}$ such that for every player m it holds that $\boldsymbol{s}_m$ is the best-response of player m with respect to player n to the pure strategy s_n.

2. Calculate the expected payoff of the strategy profile $\boldsymbol{s}$.

3. Construct a mixed strategy profile $\boldsymbol{\sigma}$ which guarantees a higher payoff than the expected payoff of the strategy profile $\boldsymbol{s}$.

Strategies Player 1 can gain information about the card on the table, through the answer of player 2. If player 1 does not ask about the card she is holding, player 1 will always learn which card is on the table. If player 1 asks about her own card, player 1 cannot know which card is lying on the table. We assume player 1 makes a correct guess about the card on the table if she knows which card is lying on the table. Furthermore we assume player 1 makes a random guess about the card on the table if she has asked about her own card. Under these assumptions, the guess of player 1 can be disregarded when analyzing the strategies of player 1, because the guess is determined by the knowledge of player 1 about the card on the table. A pure strategy of player 1

	rwb		*rww*		*rrr*		*rrw*		*rbw*	
wrb	*w*	-10	*w*	-10	*w*	4	*w*	4	*w*	4
brw	*b*	-10	*w*	0	*w*	0	*b*	4	*b*	4
rwb	*r*	-10	*r*	-10	*r*	-10	*r*	-10	*r*	-10
bwr	*b*	-10	*b*	4	*r*	0	*b*	4	*b*	4
rbw	*r*	-10	*r*	-10	*r*	-10	*r*	-10	*r*	-10
wbr	*w*	-10	*w*	4	*r*	0	*r*	0	*w*	4
E_u		-10		$-3\frac{2}{3}$		$-2\frac{2}{3}$		$-1\frac{1}{3}$		$-\frac{2}{3}$

Table 1.1: The best-response strategy of player 2 with respect to player 1.

becomes a recipe, only prescribing which question to ask, given the card player is holding.

A pure strategy of player 1 is denoted by a sequence of three characters. *bwr* denotes a strategy of player 1 such that player 1 asks about the blue card when holding the red card, she asks about the white card when holding the white card and asks about the red card when holding the blue card. The first letter represents the question asked when holding the red card, the second letter choice of question when holding the white card and the last letter denotes which question player 1 asks when holding the blue card.

The color of a card does not carry any meaning within the game Q-A. We can transpose colors without affecting the game and strategies. The pure strategy of player 1 such that she asks about the red card when holding the blue or white card and asks about the blue card when holding the red card is a permutation of the pure strategy such that player 1 asks about the white card when holding the red or blue card and asks about the blue card when holding the white card. In other words, the strategies *brr* and *wbw* are equivalent. In this example, the colors red and white are transposed. The strategies are permutations of each other. If a strategy is strictly dominated, the equivalent strategies are also strictly dominated.

We will show that every aoc-strategy is strictly dominated. We divide the set of the aoc-strategies into five sets of equivalent strategies. If one of the members of a set is strictly dominated, the whole set is strictly dominated. For every set we show that one of the members is strictly dominated.

The 19 aoc-strategies are: *rwb*, *rww*, *rwr*, *rrb*, *rbb*, *rrr*, *rrw*, *rbr*, *rbw*, *wwb*, *bwb*, *www*, *wwr*, *bww*, *bwr*, *bbb*, *wbb*, *brb*, *wrb*. We divide the set of the 19 pure aoc-strategies into five sets of equivalent strategies: {*rwb*}, {*rww, rwr, rrb, rbb, wwb, bwb*}, {*rrr, www, bbb*}, {*rrw, rbr, wwr, brr, wwb, brb*} and {*rbw, bwr, wrb*}. We will show that the strategies *rwb*, *rww*, *rrr*, *rrw* and *rbw* are strictly dominated by using the given recipe.

Best-Response First we construct the best-response strategy profiles with respect to player 1 for the five aoc-strategies. For the game Q-A, we only have to determine the best-response strategies of player 2 with respect to player 1. We have constructed these strategies with the use of table 1.1.

The head of the columns display the five aoc-strategies of player 1. The head of the rows display the six possible deals of cards. A pure strategy of player 2 is a prescript specifying which guess to make, given the deal of cards. Player

2's strategies are constrained. Player 2 cannot distinguish deal of cards if he is holding the same cards and he is asked the same question. If this is the case, player 2 has to take the same actions. We have paired the deals of cards in which player 2 is holding the same card.

On the left side of the cells, the guess of player 2, given the strategy of player 1 and the deal of cards, is displayed. The guesses are such that player 2 maximizes the number of incorrect guesses over the deal of cards. This maximizes the expected payoff of player 1. Thus, the pure strategy of player 2 determined by the table is a best-response with respect to player 1 given the strategy of player 1.

The guesses made given a strategy of player 1 construct the best-response strategy of player 2 with respect to player 1 to the strategy of player 1. For every aoc-strategy for which we want to show it is strictly dominated, we have constructed the strategy profile which maximizes player 1's payoff. The strategy profile consists of the aoc-strategy of player 1 and the strategy of player 2 determined by table 1.1 given the strategy of player 1. The second step is to calculate the expected payoff of the strategy profile for player 1.

Expected Payoff The second step is to calculate the expected payoff of the strategy profiles determined by table 1.1. For every strategy profile determined by the table we have calculated the expected payoff of player 1. Given the deal of cards, the guess of player 2 and the guessing protocol of player 1, we have calculated the expected payoff of player 1. These are displayed on the right sides of the cells. At the bottom of the columns, the expected payoff of player 1 is displayed given the strategy of player 1 and the best-response of player 2 with respect to player 1. For instance, if player 1 always asks about the red card, she uses strategy rrr, her expected payoff is $-2\frac{2}{3}$ if player 2 guesses are such that player 1's expected payoff is maximized.

Guaranteed Payoff The third step is to construct a mixed strategy which guarantees a higher expected payoff. Let the mixed strategy σ_1 be such that player 1 randomly asks about a card she is not holding. Player 1's payoff is minimized if player 2 never guesses the card on the table is the card player 1 asks him about. We distinguish two cases:

1. Player 1 asks player 2 about the card he is holding. Player 1 makes a correct guess, and player 2 guesses correct half of the times. In this case the expected payoff of player 1 is 2.

2. Player 2 asks player 2 about a card he is not holding. Player 1 makes a correct guess. Player 2 expects player 1 is not holding the card she asked about, so the this card must be lying on the table. Player 2 also makes a correct guess. The expected payoff of player 1 in this case is 0.

In the worst case, player 1's expected payoff is 1. The mixed strategy σ_1 guarantees player 1 an payoff of 1.

The guaranteed payoff of the strategy σ_1 is 1. The guaranteed payoff of σ_1 is higher than the expected payoffs of the strategy profiles determined by table 1.1. The five strategies of player 1 are strictly dominated by the strategy

σ_1. Therefore every aoc-strategy is strictly dominated. If player 1 is rational she will never ask player 2 about the card she is holding herself.

We have shown it is strictly dominated for player 1 to ask about her own card. If player 1 is rational, she will never ask about her own card, independent of the strategy of player 2. If it is common knowledge that player 1 is rational, player 2 knows player 1 never asks about the card she is holding. As soon as player 1 asks a question about a card, player 2 will deduce player 1 is not holing that card.

Nash-equilibrium Let the mixed strategy σ_1 of player 1 be such that player 1 randomly asks about a card she is not holding. Let the mixed strategy σ_2 of player 2 be such that player 2 never guesses the card on the table was the card player 1 asked about. If player 1 asks player 2 about the card he is holding, player 2 randomly picks a card he is not holding, and guesses this card is lying on the table. If player 2 is not holding the card asked about, player 2 guesses this card is lying on the table. The strategy profile $\boldsymbol{\sigma}$ is (σ_1, σ_2). We will show that the strategy profile $\boldsymbol{\sigma}$ is a Nash-equilibrium.

A Nash-equilibrium is a strategy profile such that if the players expect that the game is played according to the Nash-equilibrium, the players will not deviate form their strategy. We will show that if player 1 uses strategy σ_1, if player 2 uses uses strategy σ_2, and vice versa.

Suppose player 1 uses strategy σ_1. Player 2 does not expect player 1 to ask about her own card. Player 1 randomly asks about a card she is not holding. Player 2 will not deviate from his above described guessing protocol, because this will not increase his payoff.

Suppose player 2 uses strategy σ_2. Player 1 is dealt a card. Suppose player 1 changes her mind and asks player 2 about the card she is holding. If she does so, she will guess incorrectly in half of the times. Player 2 will always make an incorrect guess. If player 1 makes an incorrect guess her payoff is -24, if she makes an correct guess, her payoff is 8. Player 1's expected payoff is 8 - 24 = -16. If player 1 sticks to her strategy of the strategy profile, we distinguish three cases:

1. Half of the times player 1 asks about a card player 2 is not holding. Both players make a correct guess. The payoff of player 1 is 0.

2. Quarter of times player 1 asks about the card player 2 is holding and player 2 makes an incorrect guess. The payoff of player 1 is 4.

3. In the last quarter of the times, player 1 asks about the card player 2 is holding and player 2 makes a correct guess. The payoff of player 1 is 0.

Player 1's expected payoff is $\frac{1}{2} \times 0 + \frac{1}{4} \times 4 + \frac{1}{4} \times 0 = 1$. Therefore, player 1 will not deviate from her strategy according to the strategy profile. The strategy profile (σ_1, σ_2) is a Nash-equilibrium.

1.3.3 Knowledge

In this subsection, we describe how players develop definite and strategic knowledge in the game Q-A.

Definite Knowledge Definite knowledge is the knowledge players develop, because of explicit information exchange. In the game Q-A, player 1 and player 2 develop definite knowledge when player 2 answers the question of player 1. In this paragraph we describe the definite knowledge the players develop in the game Q-A.

When player 1 asks player 2 about the card player 2 is holding, player 1 learns which card is lying on the table. In other words, player 1 knows the deal of cards. After the answer, player 2 knows player 1 knows the distribution of cards. If player 2 does not have the card player 1 asks about, player 2 does not know whether player 1 knows the deal of cards. This depends on the card of player 1. If player 1 asks player 2 about the card she is holding herself, both players do not learn anything about the card on the table. If player 1 asks about a card both players do not hold, this card must be lying on the table, and only player 1 knows this after the answer of player 2. How the players develop definite knowledge is also common knowledge.

Strategic Knowledge Players develop strategic knowledge if the players have knowledge about the used strategies. In this paragraph we give examples of how strategic knowledge could be developed in the game Q-A.

Always ask your own card If player 1 always asks about her own card, she uses strategy *rwb*. Suppose it is common knowledge this strategy is used. Now, player 1 asks her question. As soon as player 1 has asked her question, player 1 and player 2 commonly know player 1 is holding the card she asked player 2 about. Furthermore, player 2 learns which card is lying on the table. So, the question of player 1 will inform player 2 about the card on the table.

Always ask the red card Suppose player 1 always asks about the red card and it is common knowledge that she uses this strategy. The strategy is denoted by *rrr*. Player 1 asks player 2 her question "Are you holding the red card ?". The players do not develop any knowledge about the deal of cards. The players already knew player 1 would ask about the red card.

Never ask your own card Player 1 uses the strategy *wbw*. Player 1 asks about the white card when holding the red or blue card. She asks about the blue card when holding the white card. It is common knowledge player 1 uses this strategy.

Player 1 asks player 2 about the white card. Player 2 learns player 1 must be holding the red or blue card. If player 2 is holding the white card, he does not gain any information. If player 2 is holding the blue or red himself, he knows which card is lying on the table after the question of player 1. If player 1 asks player 2 about the blue card, player 2 learns player 1 is holding the white card.

Strict domination It is common knowledge player 1 is rational. Rational players do not use strictly dominated strategies. Therefore it is common knowledge player 1 will never ask about her own card. The question is asked. Player 2 learns player 1 is not holding the card player 1 asks about, because he knows player 1 is rational. If player 2 is also not holding the card player 1 asks about, player 2 knows which card is lying on the table. So, player 1 asks her question to learn about the card on the table, but what really happens when player 1 asks her question is that player 2 learns before player 1 what the card on the table is. We call this specific form of strategic knowledge *rational knowledge*, because it is developed if it is common knowledge when players are rational.

Nash-equilibrium Suppose it is common knowledge Q-A is played according to the Nash-equilibrium $\boldsymbol{\sigma}$, described in the previous subsection. Player 1 never asks about her own card. When player 1 asks her question, player 2 learns player 1 is not holding the card player 1 is asking player 2 about. Player 2 uses this information to make his guess about the card on the table. If player 2 is not holding the card player 1 asked him about, player 2 guesses the card on the table is the card player 1 asked about.

Uncommon knowledge Suppose player 1 uses strategy rrw. Player 1 asks about the white card when holding the blue card. Otherwise, player 1 asks about the red card. Through some event, player 2 has learned player 1 will use this strategy. Player 1 is not informed about player 2's knowledge. Player 1 expects player 2 not to have any knowledge about the strategy player 1 uses.

Player 1 asks player 2 about the red card. After the question, player 2 privately knows player 1 is holding the red or white card. If player 2 is holding one of these cards, player 2 knows which card is lying on the table. Player 1 does not know and expect that player 2 can know the card on the table.

In another case, player 1 asks player 2 about the white card. Both players now know player 1 is holding the blue card, but player 1 does not know that player 2 has this knowledge. On the other hand, player 2 knows player 1 knows she is holding the blue card.

All these examples are examples of how players can develop strategic knowledge in the game Q-A. In the game Q-A, and in general, strategic knowledge is used to determine the actions to take during the game.

1.3.4 Conclusions of a Card Game

In this section we have analyzed a card game called Q-A. We have determined what the available strategies to the players are. A strategy of player 1 is a recipe prescribing for every circumstance which question to ask and guess to make. The strategy of player 2 tells him which guess to make, given the question of player 1 and the card player 2 is holding.

If player 1 is rational, she will never ask about her own card. The action of asking about your own card is strictly dominated. A Nash-equilibrium of the game is a strategy profile such that player 1 randomly asks player 2 about a card she is not holding. Player 2 uses this information in such a way that player 2 guesses the card on the table is the card player 1 asked about if player 2 is not holding this card.

We have given several examples of development of strategic knowledge in the game Q-A. For instance, if it is common knowledge players are rational, player 1 never asks about the card she is holding. In this case, when player 1 asks her question, player 2 learns which card player 1 is not holding. Furthermore we have described how players develop definite knowledge during the game.

1.4 Conclusions

In this chapter we have given an introduction to game theory and rational behavior. Given the game theoretic tools, we can precisely describe what game, definite, and strategic knowledge is in a game of imperfect information.

For a small card game, called Q-A, we have analyzed what the definite and strategic knowledge is players can develop the game. Furthermore, we have given several examples of strategic knowledge development in the game.

The goal of this thesis is to model definite and strategic knowledge and their development in games of imperfect information. In this chapter we have given a precise description of what is understood as definite and strategic knowledge in games of imperfect information. Rests us to come up with a satisfying model of definite and strategic knowledge and its development in games of imperfect information.

Chapter 2

Modeling Knowledge

2.1 Introduction

The object in view is to model definite and strategic knowledge and its development in games of imperfect information. We find knowledge in general naturally modeled by Kripke models. Given a set of agents, a Kripke model consists of a set of possible states and accessibility relation between the states for every agent. States which are accessible form each other for an agent are indistinguishable for the agent. For games of imperfect information, we can model the knowledge of the players about the state of the game by viewing game states as Kripke state and players as agents.

Players develop knowledge in a game of imperfect information when actions are taken. We call actions which alter the knowledge of the players knowledge actions. Knowledge development can be modeled by modeling the knowledge actions as action models. An action model is a Kripke model modeling knowledge actions. Given an initial Kripke model modeling the knowledge of players, the action models can be sequentially executed in the Kripke model, resulting in a new Kripke model, modeling the knowledge of the players after the action model.

Kripke models and action models are semantical structures of dynamic epistemic logic. In this chapter we introduce a dynamic epistemic logic. Given the dynamic epistemic logic, we will analyze and model the definite and strategic knowledge and its development in the card game Q-A. Unfortunately we have to conclude that strategic knowledge cannot be satisfyingly modeled by dynamic epistemic logic.

2.2 Dynamic Epistemic Logic

In this section we introduce epistemic and dynamic epistemic logic. For further reading on epistemic logic, we refer to Meyer and van der Hoek's *Epistemic Logic for AI and Computer Science* [16].

2.2.1 Epistemic Logic

First we introduce epistemic logic. Epistemic logic can be used to model knowledge in games of imperfect information. In epistemic logic knowledge is attributed to agents.

Agents An agent is an identity to which knowledge can be attributed. Agents are assumed to be perfect logicians, i.e. the agents know all the consequences of their knowledge. When modeling knowledge development in games, players are modeled as agents. We thus assume players are perfect logicians.

Epistemic Language An epistemic language is a propositional language extended with knowledge operators. In the epistemic language we can construct sentences describing the knowledge of the agents.

Definition 10 (Epistemic Language)
Let N be a finite set of agents and $\mathcal{P}$ be a finite set of propositional atoms. The *epistemic language* $\mathcal{L}_{\mathcal{P},N}$ is the smallest closed set for which holds:

- $p \in \mathcal{P} \Rightarrow p \in \mathcal{L}_{\mathcal{P},N}$;
- $\phi, \psi \in \mathcal{L}_{\mathcal{P},N} \Rightarrow \neg\phi, (\phi \wedge \psi) \in \mathcal{L}_{\mathcal{P},N}$;
- $\phi \in \mathcal{L}_{\mathcal{P},N}$ and $n \in N \Rightarrow K_n\phi \in \mathcal{L}_{\mathcal{P},N}$.

The sentence $K_1\phi$ is read as: agent 1 knows ϕ holds. We can construct formulas with several K-operators such as $K_1\neg K_2 r_1$. Let r_1 mean agent 1 is holding the red card, then the sentence is read as: agent 1 knows agent 2 does not know agent 1 is holding the red card. Let $p \in \mathcal{P}$, then $(\phi \vee \psi)$, $\phi \rightarrow \psi$, $\phi \leftrightarrow \psi$ and $\top$ are abbreviations for $\neg(\neg\phi \wedge \neg\psi)$, $\neg\psi \vee \phi$, $(\psi \wedge \phi) \vee (\neg\psi \wedge \neg\phi)$ and $\neg(p \wedge \neg p)$ respectively.

Kripke model A Kripke model consists of a set of states and accessibility relations between the states for every agent. Every state is viewed as an possible state of the world, of which one is the actual state. If states are accessible from each other for an agent, the agent cannot distinguish the states. An agent knows which is the actual state, if no other state than the actual state can be accessed from that state.

Definition 11 (Kripke Model)
Let a finite set of agents N and a finite set of propositional atoms $\mathcal{P}$ be given. A *Kripke model* M is a tuple (W, R, V):

- The set W is a nonempty set of states $\{w_1, \ldots, w_{|W|}\}$;
- The *accessibility function* $R : N \rightarrow 2^{W \times W}$ is a function which assigns to each agent a set of ordered pairs of states. For every world w, w', w'' and agent n holds that: $(w, w) \in R(n)$ (reflexivity). If $(w, w') \in R(n)$ then $(w', w) \in R(n)$ (symmetry). If $(w, w'), (w', w'') \in R(n)$ then $(w, w'') \in R(n)$(transitivity);
- The *valuation function* $V : W \rightarrow 2^{\mathcal{P}}$ assigns to each state a set of propositional atoms.

A Kripke world is a pair consisting of a Kripke model $M = (W, R, V)$ and a state $w \in W$ and is denoted by (M, w).

$(w, w') \in R(n)$ is interpreted as: state w' is accessible from state w for agent n. The accessibility relations are reflexive, transitive and symmetrical, i.e. R defines an equivalence relation. The set of propositional atoms assigned to a state by V are the atoms which hold in that state.

Kripke models can be used to model knowledge of players. The states of a game are modeled as Kripke states. The accessibility relations are defined corresponding to the indistinguishability of the states to the players.

Semantics Given a Kripke model, modeling the knowledge distribution over the agents, an epistemic sentence can be tested in a Kripke world. To test a sentence is to determine its truth value. The truth value is determined by the semantics of the epistemic language.

Definition 12 (Semantics for $\mathcal{L}_{\mathcal{P},\mathcal{N}}$)
Let a Kripke model $M = (W, R, V)$ and the epistemic language $\mathcal{L}_{\mathcal{P},N}$ be given.

$$\begin{array}{lcl} M, w \models p & \Leftrightarrow & p \in V(w); \\ M, w \models \neg\phi & \Leftrightarrow & M, w \not\models \phi; \\ M, w \models \phi \wedge \psi & \Leftrightarrow & M, w \models \phi \text{ and } M, w \models \psi; \\ M, w \models K_n\phi & \Leftrightarrow & \text{For all worlds } w' \text{ such that } (w, w') \in R(n) \text{ holds: } M, w' \models \phi. \end{array}$$

The accessibility relations of a Kripke model are constrained. The relations are necessarily reflexive, transitive and symmetrical. These three constraints restrict the knowledge agents can have. Each constraint corresponds to one restriction. Because the accessibility relations are reflexive known facts are true, i.e. $K_n\phi \rightarrow \phi$. Agents cannot have incorrect knowledge. As the relations are transitive the agents know what they know, i.e. $K_n\phi \rightarrow K_nK_n\phi$. Symmetry induces that agents have complete knowledge about what they do not know, i.e. $\neg K_n\phi \rightarrow K_n\neg K_n\phi$. When modeling knowledge of players by Kripke models, the knowledge of the players is also restricted by the constraints of the accessibility relations.

2.2.2 Dynamic Epistemic Logic

Kripke models only model the knowledge of agents at a certain moment in time. Some events may occur which alter the knowledge of the players. In that case the Kripke model needs to be updated to correctly model the knowledge of the agents. Events which alter the knowledge of the agents are called knowledge actions. A knowledge action is modeled by an action model, modeling the knowledge development of the agents, induced by the knowledge action. The action models are used to update Kripke models.

Actions Model Actions models are very similar to Kripke models. An action model consists of a set of actions and an accessibility relation between the actions for every agent.

Definition 13 (Action Model)
Let a set of agents N and the epistemic language $\mathcal{L}_{\mathcal{P},N}$ be given. An *action model* μ is a tuple (A, R, pre):

- The set A is a nonempty set of actions $\{a_1, \ldots, a_{|A|}\}$;
- The *accessibility function* $R : N \to 2^{A \times A}$ is a function which assigns to each agent a set of ordered pairs of actions. For every action a, a', a'' and agent n holds that: $(a, a) \in R(n)$ (reflexivity). If $(a, a') \in R(n)$ then $(a', a) \in R(n)$ (symmetry). If $(a, a'), (a', a'') \in R(n)$ then $(a, a'') \in R(n)$(transitivity);
- The *precondition function pre* : $A \to \mathcal{L}_{\mathcal{P},\mathcal{N}}$ assigns to every action a precondition.

Action models model events which alter the knowledge of agents. The set of actions can be viewed as several possible events, of which one actually occurs. On the set of actions the accessibility function is defined. Actions are accessible from each other for an agent if the agent cannot distinguish the action. If an action of the set of actions takes place, some agents can be ignorant which action actually takes places, as they cannot distinguish an action from another actions. Which actions the agents cannot distinguish from each other is defined by the accessibility relation.

In a game of imperfect information, knowledge actions which occur during the game are modeled by action models. For instance the answer to a question about the deal of cards can be modeled by and action model. Every possible answer to the question is viewed as an separate actions.

Execution Action models are used to update Kripke model. This is done by the execution of every action of the action model in every state of the Kripke model. The execution of an action in a state results in a new state if and only if the precondition of the action holds in the state. The execution of the action in the states results in a new Kripke model. An action model is an operator on a Kripke model. The execution can be seen as a multiplication of the action model with the Kripke model.

Definition 14 (Execution)
Let a Kripke model $M = (W, R, V)$ and an action model $\mu = (A, R, pre)$ be given. The execution of action model μ in Kripke model M results in a Kripke model denoted by $M \otimes \mu$. $M \otimes \mu = (W', R', V')$ such that:

- The set of worlds is $W' = \{(w, a) \in W \times A | M, w \models pre(a)\}$;
- The accessibility function R' is such that for every player n holds $((w, a), (w', a')) \in R'(n)$ iff $(w, w') \in R(n)$ and $(a, a') \in R(n)$.
- The valuation function V' is such that $V'(w, a) = V(w)$.

The resulting set of states is the set of pairs of states and actions for which holds that the precondition of the the action holds in the state. On the new set of states, accessibility is defined. A pair of a state and an action (w, a) is accessible from another state and action pair (w', a') for an agent if the states and the actions are accessible from each other for the agent. The valuation of a state is unchanged.

In games of imperfect information, knowledge and its development can be modeled by Kripke models, action models and execution. The knowledge of the players about the state of the game at the beginning of the game is modeled by a Kripke model. The knowledge actions which occur during the game are modeled by action models. The knowledge development is modeled by sequential execution of the action models in the Kripke model, resulting in a new up-to-date Kripke model modeling the knowledge of the players after the knowledge actions. The knowledge of the players can be tested with the use of the semantics of the epistemic language in which the knowledge of the players is described.

2.2.3 Example Muddy Children Puzzle

As an example and application of our dynamic epistemic logic, we analyze a version of the muddy children problem with dynamic epistemic logic. Other analyses the problem are given by Fagin, Halpern, Moses & Vardi [9], Meyer & van der Hoek [16], Gerbrandy & Groeneveld [10] and Baltag [1].

The Puzzle The puzzle is about a father and 2 children. During play, both children have obtained muddy foreheads. No child can see his own forehead, only the forehead of the other child. Along comes father and he utters: "At least one of you has a muddy forehead!". Both children already know this, as they can see each other's forehead. Then father asks child 1: "Can you prove you have mud on your forehead?" and child 1 answers "No". Then father asks child 2 whether he can prove is forehead is muddy. What will child 2 answer ?

The following is assumed to be common knowledge: 1) The children are prefect logicians 2) The children never lie. When child 1 answers father's question of father, it becomes commonly known that child 1 does not know his forehead is muddy. If child 2's forehead would not have been muddy, child 1 would know his forehead is muddy, as at least one of the children is muddy. But child 2's forehead is muddy and child 1 holds two alternatives as possible: 1) Child 1's forehead is clean and 2) Child 1's forehead is muddy. Child 2 hears the answer of child 1. He deduces it must the case his forehead is muddy, otherwise child 1 would have said he knew his forehead was muddy. And so child 2 answers "Yes, I can prove my forehead is muddy.".

The knowledge, knowledge development and reasoning involved in the muddy children puzzle can be modeled by dynamic epistemic logic. The initial knowledge of the children about the state of their foreheads is modeled by a Kripke model. The utterance of father and the responses of the children to the questions are knowledge actions, modeled by action models. The knowledge the children develop is calculated by the sequential execution of the action models.

Let the set of agents N consist of child 1 and child 2, i.e. $\{1, 2\}$, and the set of propositional atoms $\mathcal{P}$ be $\{m_1, m_2\}$. m_1 and m_2 respectively mean child 1's forehead is muddy and child 2's forehead is muddy.

Kripke Model The initial knowledge distribution over the children is modeled by Kripke model $M = (W, R, V)$:

- $W = \{\mathsf{cc}, \mathsf{mc}, \mathsf{cm}, \mathsf{mm}\}$.
- $R(1) = \{(\mathsf{cc}, \mathsf{cc}), (\mathsf{mc}, \mathsf{mc}), (\mathsf{cm}, \mathsf{cm}), (\mathsf{mm}, \mathsf{mm}), (\mathsf{cc}, \mathsf{mc}), (\mathsf{mc}, \mathsf{cc}), (\mathsf{mm}, \mathsf{cm}), (\mathsf{cm}, \mathsf{mm})\}$ and $R(2) = \{(\mathsf{cc}, \mathsf{cc}), (\mathsf{mc}, \mathsf{mc}), (\mathsf{cm}, \mathsf{cm}), (\mathsf{mm}, \mathsf{mm}), (\mathsf{cc}, \mathsf{cm}), (\mathsf{cm}, \mathsf{cc}), (\mathsf{mc}, \mathsf{mm}), (\mathsf{mm}, \mathsf{mc})\}$.

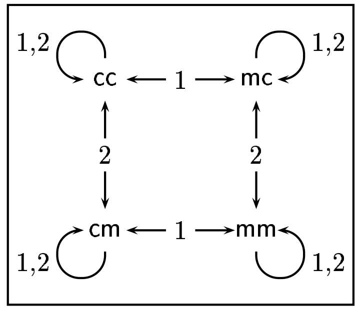

Figure 2.1: The Kripke model for the muddy children puzzle.

- $V(\mathsf{cc}) = \emptyset$, $V(\mathsf{mc}) = \{m_1\}$, $V(\mathsf{cm}) = \{m_2\}$, $V(\mathsf{mm}) = \{m_1, m_2\}$.

The Kripke model is represented by a graph. The nodes are the states of the Kripke model. The nodes are labeled with the names of the states. An accessibility relation for agent between state corresponds a connection between the states labeled with the name of the agent. Figure 2.1 is the graphical representation of the Kripke model M.

A state of the Kripke model describes which children are muddy and which are not. mc is the state in which child 1 is muddy and child 2 is clean. States are accessible from each other for a child if in the states the conditions of the other child's forehead correspond. For instance, state mc is accessible from state mm for child 2, because the forehead of child 1 is muddy in both states.

Action Models The knowledge development induced by father's announcement and the answer of the child are modeled by action models. The action model *father* models the utterance of father and is such that $father = (A, R, pre)$:

- $A = \{\textbf{muddy}\}$;
- $R(1) = R(2) = \{(\textbf{muddy}, \textbf{muddy})\}$;
- $pre(\textbf{muddy}) = m_1 \vee m_2$.

The action model consists of only one action. The names of the actions are chosen such that the action models are readable. The actions itself do not carry any meaning. The precondition states that at least one of the children's foreheads muddy is.

The answer "No" of child 1 to the question "Can you prove your forehead is muddy?" is modeled by the action model $no = (A, R, pre)$:

- $A = \{\textbf{dontknow}\}$;
- $R(1) = R(2) = \{(\textbf{dontknow}, \textbf{dontknow})\}$;
- $pre(\textbf{dontknow}) = \neg K_1 m_1$.

The precondition states that child 1 does not know that his forehead is muddy, as child 1 cannot prove his forehead is muddy.

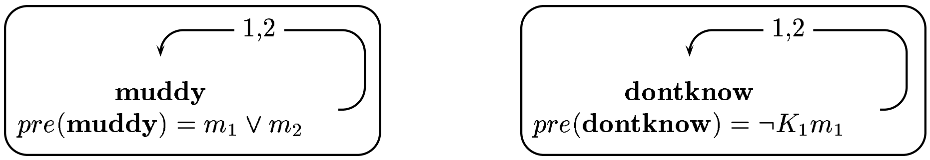

Figure 2.2: The action models for the muddy children puzzle.

Just as Kripke models, we display the action models graphicly. The action models are displayed in figure 2.2. Both action models are simple action models, because they both only consist of only one action.

Execution Execution of the action models in the Kripke model corresponds to the knowledge development and reasoning of the children in the muddy children problem. The execution of the action model $father$ in Kripke model M results in Kripke model $M \otimes father = M' = (W', R', V')$:

- $W' = \{(\mathsf{cm}, \mathbf{muddy}), (\mathsf{mc}, \mathbf{muddy}), (\mathsf{mm}, \mathbf{muddy})\}$;
- $R'(1) = \{((\mathsf{cm}, \mathbf{muddy}), (\mathsf{cm}, \mathbf{muddy})), ((\mathsf{mc}, \mathbf{muddy}), (\mathsf{mc}, \mathbf{muddy})), ((\mathsf{mc}, \mathbf{muddy}), (\mathsf{mm}, \mathbf{muddy})), ((\mathsf{mm}, \mathbf{muddy}), (\mathsf{mc}, \mathbf{muddy})), ((\mathsf{mm}, \mathbf{muddy}), (\mathsf{mm}, \mathbf{muddy}))\}$ and $R'(2) = \{((\mathsf{cm}, \mathbf{muddy}), (\mathsf{cm}, \mathbf{muddy})), ((\mathsf{cm}, \mathbf{muddy}), (\mathsf{mm}, \mathbf{muddy})), ((\mathsf{mm}, \mathbf{muddy}), (\mathsf{cm}, \mathbf{muddy})), ((\mathsf{mc}, \mathbf{muddy}), (\mathsf{mc}, \mathbf{muddy})), ((\mathsf{mm}, \mathbf{muddy}), (\mathsf{mm}, \mathbf{muddy}))\}$;
- $V'(\mathsf{cm}, \mathbf{muddy}) = \{m_2\}$, $V'(\mathsf{mc}, \mathbf{muddy}) = \{m_1\}$ and $V'(\mathsf{mm}, \mathbf{muddy}) = \{m_1, m_2\}$.

The execution of action model no in Kripke model M' results in Kripke model $M' \otimes no = M'' = (W'', R'', V'')$:

- $W'' = \{(\mathsf{mc}, \mathbf{muddy}, \mathbf{dontknow}), (\mathsf{mm}, \mathbf{muddy}, \mathbf{dontknow})\}$;
- $R''(1) = \{((\mathsf{mc}, \mathbf{muddy}, \mathbf{dontknow}), (\mathsf{mc}, \mathbf{muddy}, \mathbf{dontknow})), ((\mathsf{mc}, \mathbf{muddy}, \mathbf{dontknow}), (\mathsf{mm}, \mathbf{muddy}, \mathbf{dontknow})), ((\mathsf{mm}, \mathbf{muddy}, \mathbf{dontknow}), (\mathsf{mc}, \mathbf{muddy}, \mathbf{dontknow})), ((\mathsf{mm}, \mathbf{muddy}, \mathbf{dontknow}), (\mathsf{mm}, \mathbf{muddy}, \mathbf{dontknow}))\}$ and $R''(2) = \{((\mathsf{mc}, \mathbf{muddy}, \mathbf{dontknow}), (\mathsf{mc}, \mathbf{muddy}, \mathbf{dontknow})), ((\mathsf{mm}, \mathbf{muddy}, \mathbf{dontknow}), (\mathsf{mm}, \mathbf{muddy}, \mathbf{dontknow}))\}$;
- $V''(\mathsf{mc}, \mathbf{muddy}, \mathbf{dontknow}) = \{m_1\}$ and $V''(\mathsf{mm}, \mathbf{muddy}, \mathbf{dontknow}) = \{m_1, m_2\}$.

Figure 2.3 graphically displays the sequential execution of the action models $father$ and no in Kripke model M. The figure displays the Kripke models M' and M'' which result from the execution of the action models in Kripke model M. Within the Kripke models M' and M'', the names of the states are abbreviations for the sake of convenience and readability. Also the accessibility relations of the action models are left out.

As can be seen in figure 2.3, the execution of the action model $father$ results in the deletion of state cc. No child holds the state as possible anymore in any

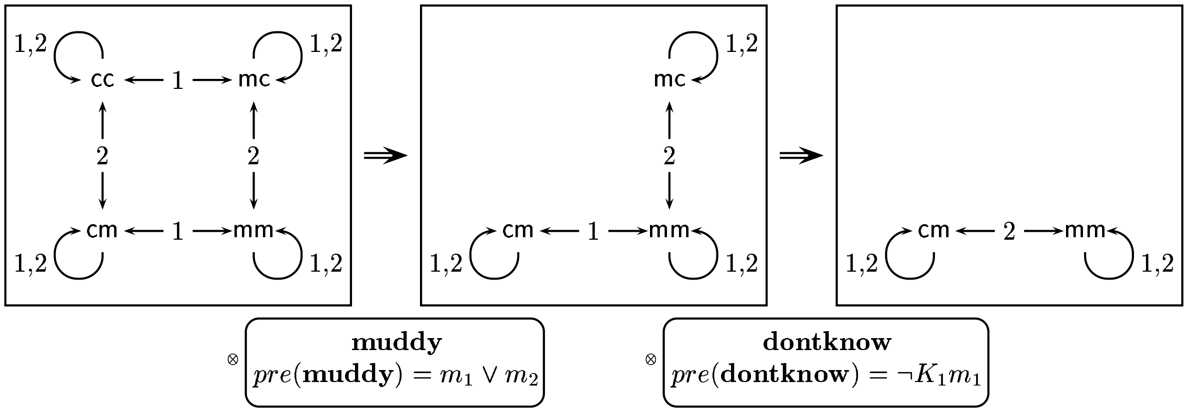

Figure 2.3: Knowledge development in the muddy children puzzle

of the states. It has become common knowledge that at least one of the children is muddy.

The answer of child 1 results in the deletion of state cm. Child 2 now knows his forehead is muddy. This corresponds to the sentence $K_2 m_2$. The world consisting of state mm is the state in which both children have mud on their foreheads. Sentence $K_2 m_2$ holds in world $(M'', (\mathsf{mm}, \textbf{muddy}, \textbf{dontknow}))$, as the the sentence m_2 holds in every world accessible for child 2 from world $(M'', (\mathsf{mm}, \textbf{muddy}, \textbf{dontknow}))$. The accessible worlds for child 2 are again $(M'', (\mathsf{mm}, \textbf{muddy}, \textbf{dontknow}))$. In this world it holds that child 2's forehead is muddy.

If both children have mud on their foreheads, after utterance of father and the answer of child 1, child 2 knows his forehead is muddy. We showed this by analyzing the puzzle with dynamic epistemic logic. With the use of dynamic epistemic logic, the knowledge development of the children can automatically and elegantly be calculated by the execution of the action models in the initial Kripke model.

2.2.4 Baltag

Our action model is an instance of to the action-model defined by Baltag in *A Logic for Suspicious Players* [1]. We constrain the accessibility function of the action model to equivalence relations. Baltag's action-model's accessibility relation on the other hand, defines not necessarily a equivalence relation. Baltag's action-models are semantical structures for the semantics of a dynamic epistemic language. The dynamic epistemic language is an epistemic language extended with action-expressions. An action-expression describes an event and the appearance of the event to the agents. For instance, the utterance of father "At least one of you has mud on your forehead" is described by the action-expression $(?(m_1 \vee m_2))^{*\{1,2\}}$. The group $\{1, 2\}$ commonly learns the sentence $m_1 \vee m_2$ holds. The action expression is added to the epistemic language as an operator. The sentence $[\alpha]\phi$ containing the action expression α is read as: after action α sentence ϕ holds. In our logic, actions are not described, only modeled.

Every action-expression describing an event corresponds to an action-model which models the event. This correspondence is defined by a function. The corresponding action model of action-expression α is $\bar{\alpha}$. Let action model $\bar{\alpha}$ consist of the set of actions A. The semantics of an action-expressions is such that sentence $[\alpha]\phi$ holds in Kripke world (M, w) iff for every action $a \in A$ holds $(M \otimes \bar{\alpha}, (w, a)) \models \phi$. So after action α, sentence ϕ holds if ϕ holds in all worlds resulting from the execution of the corresponding action-model $\bar{\alpha}$ of α.

2.2.5 Conclusions Dynamic Epistemic Logic

We have introduced a dynamic epistemic logic similar to Baltag's logic presented in *A Logic for Suspicious Players* [1]. The logic is also inspired by the logic introduced by Van Ditmarsch' logic presented in *Knowledge Games* [20] and the logic introduced by Kooi, Van Ditmarsch and Van der Hoek in *Action Language* [15]. The logic we present in this master thesis can be seen as an instance of Baltag's logic. We do not provide an axiomatization of the logic as it is beyond the scope of this master thesis.

We have given an analysis of a version of the muddy children puzzle which showed how the dynamic epistemic logic can be used to model knowledge development.

2.3 Analysis of a Card Game

In chapter 1 we introduced the game Q-A. Q-A is a card game consisting of two players and three cards. We described how the players develop definite knowledge and how the players can develop strategic knowledge. In this section we analyze the knowledge development of the players with dynamic epistemic logic.

2.3.1 Game & Kripke Model

In this subsection we present the description of the game and we go into the modeling of the knowledge of the players. Player 1 and 2 will be addressed with she and he respectively.

Description of Q-A The game consists of two players, a table and three cards. The cards are a red, a blue and a white card. Both players are dealt a card and the third card is put on the table, facing down. The players can only see their own card.

After the dealing of the cards, player 1 asks player 2 a question. Player 1 has a choice of three possible questions: 1. "Are you holding the red card?"; 2. "Are you holding the white card?" and 3. "Are you holding the blue card?". Player 1 is the only player to ask a question. Then, player 2 answers player 1's question. Player 2 must answer the question of player 1 truthfully. When player 2 has answered player 1's question, both players make a guess about the card on the table and the card on the table is shown to the players. Correctly guessing players are rewarded, while incorrectly guessing players are punished.

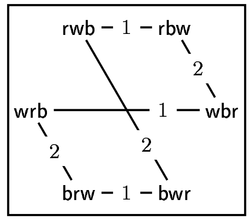

Figure 2.4: Kripke model of the game Q-A

Kripke model of Q-A The initial state of the game is determined by the distribution of the cards over the players. The states of the Kripke model, modeling the initial knowledge of the players are the possible deals of cards.

A deal of cards is denoted by a sequence of letters. rwb denotes player 1 is holding the red card, player 2 is holding the white card and the blue card is lying on the table. Six deals of cards are possible, namely rwb, rbw, wrb, wbr, brw, bwr. Players cannot distinguish deals of cards in which they are holding the same card. So deal rwb cannot be distinguished from bwr by player 2, because he is holding the white card in both deals of cards.

The players reason about who is holding which card, and which card is lying on the table. The propositional atoms describe which player is holding what card. For instance, the atoms w_2 means player 2 is holding the white card. The other propositional atoms have corresponding meanings. The set of propositional atoms is $\{r_1, w_1, b_1, r_2, w_2, b_2, r_t, w_t, b_t\}$. The propositions r_t, w_t and b_t denote which card is lying on the table. The set of agents consists of the players 1 and 2, i.e. $\{1, 2\}$. The Kripke model is $M = (W, R, V)$:

- $W = \{\mathsf{rwb}, \mathsf{rbw}, \mathsf{wrb}, \mathsf{wbr}, \mathsf{brw}, \mathsf{bwr}\}$;
- $R(1) = \{(\mathsf{rwb}, \mathsf{rbw}), (\mathsf{wrb}, \mathsf{wbr}), (\mathsf{brw}, \mathsf{bwr})\}$ and $R(2) = \{(\mathsf{wrb}, \mathsf{brw}), (\mathsf{rwb}, \mathsf{bwr}), (\mathsf{rbw}, \mathsf{wbr})\}$;
- $V(\mathsf{rwb}) = \{r_1, w_2, b_t\}$, $V(\mathsf{rbw}) = \{r_1, b_2, w_t\}$, $V(\mathsf{wrb}) = \{w_1, r_2, b_t\}$, $V(\mathsf{wbr}) = \{w_1, b_2, r_t\}$, $V(\mathsf{wbr}) = \{w_1, b_2, r_t\}$, $V(\mathsf{brw}) = \{b_1, r_2, w_t\}$ and $V(\mathsf{bwr}) = \{b_1, w_2, r_t\}$.

The Kripke model is displayed in figure 2.4. The accessibility relations are reflexive, transitive and symmetrical. For reasons of readability we have left out reflexive accessibility relations from the definition and figure of the Kripke model. Furthermore the accessibility relation is a set of unordered pairs, instead of ordered pairs, as the accessibility relation is symmetrical. In the figure, a connection between two states represents that the states are accessible from each other.

2.3.2 Definite Knowledge

During the game, the players develop definite knowledge when player 2 answers the question of player 1. Suppose player 1 asks player 2 about the red card, and player 2 answers the question. We have modeled the knowledge development induced by the question of player 1 in dynamic epistemic logic.

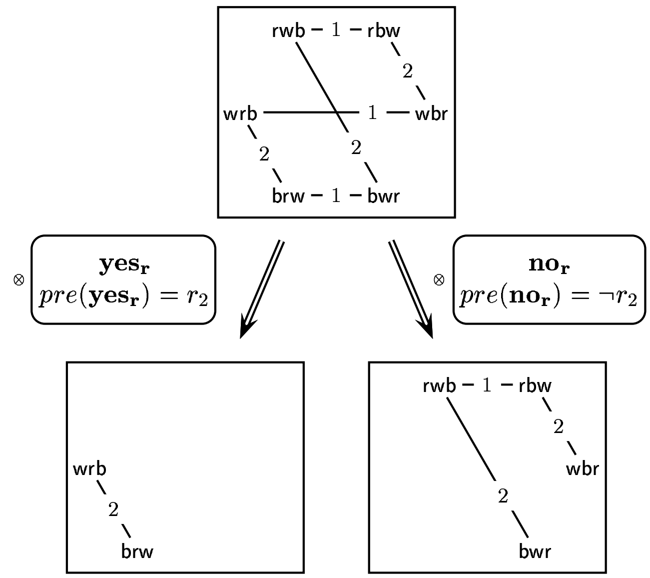

Figure 2.5: Definite knowledge development in Q-A

Suppose player 2 answers affirmative, he is holding the red card. The knowledge the players develop is modeled by the action model $yes_r = (A, R, pre)$:

- $A = \{\mathbf{yes}\}$;
- $R(1) = R(2) = \emptyset$;
- $pre(\mathbf{yes}) = r_2$.

The action model only contains one action, thus the accessibility relation is trivial. The precondition states player 2 is holding the red card. If player 2 is not holding the red card, player 2 informs player 1 that he is not holding the red card. The action model is $no_r = (A, R, pre)$:

- $A = \{\mathbf{no}\}$;
- $R(1) = R(2) = \emptyset$;
- $pre(\mathbf{no}) = \neg r_2$.

Figure 2.5 displays the execution of the action models in the Kripke model.

If player 2 is holding the red card, player 1 knows the distribution of the cards, because player 1 can distinguish every deal of cards. Player 2 does not know whether the blue or white card is lying on the table. Player 2 does know player 1 knows the distribution of the cards, because in every state player 1 knows the deal of cards.

Suppose player 2 is not holding the red card. Let the actual deal of cards be bwr. Player 1 knows the distribution of the cards, because no deal of cards is accessible for player 1 from the deal bwr. Player 2 holds two deals of cards for possible, bwr and rwb. Furthermore player 2 does not know whether player

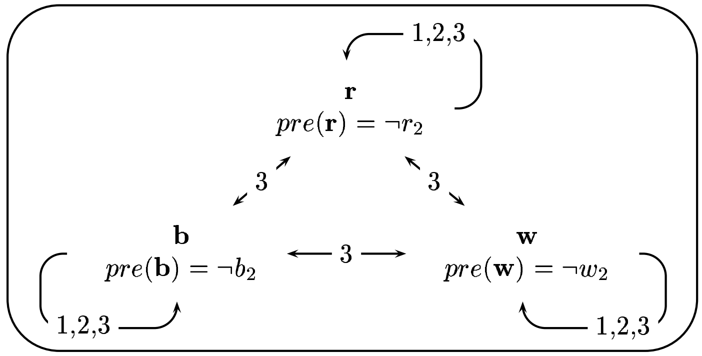

Figure 2.6: Action model *show*

1 knows the distribution of cards, because if the deal of cards is rwb, player 1 does not know the deal of cards. Now suppose rwb is the actual deal of cards. In this player 1 has asked about her own card. After the question and answer, player 1 holds the deals rwb and rbw as possible and player 2 holds the deals rwb and bwr as possible and both players do not know which card is lying on the table.

Example Whispering The action models presented in the game Q-A are simple action models. The action models only consisted of one action. Here we will give an example in which the action model will contain more actions.

The example is about a card game very similar to Q-A. In our example, three cards are dealt over three players. The cards are a red, a white and a blue card. The players are called 1, 2 and 3. The players can only see their own card. A Kripke model modeling the the knowledge distribution in this would consists of the six possible deals of cards and accessibility relations defined between the deals of cards.

After the dealing of the cards player 1 asks player 2 which card he is not holding. Player 2 can give two answers. For instance, if player 2 is holding the white card, he can answer "I am not holding the blue card" and "I am not holding the red card". The answer of player 2 is not made publicly. Player 2 whispers his answer in player 1's ear. Player 3 does not know which answer player 2 has given.

Let r_n, w_n, b_n respectively mean player n is holding the red card, player n is holding the white card and player n is holding the blue card. Given these propositional atoms we model the knowledge action described above as the action model $show = (A, R, pre)$:

- $A = \{\mathbf{r}, \mathbf{w}, \mathbf{b});$
- $R(1) = R(2) = \emptyset$ and $R(3) = \{(\mathbf{r}, \mathbf{w}), (\mathbf{r}, \mathbf{b}), (\mathbf{w}, \mathbf{b})\};$
- $pre(\mathbf{r}) = r_2$, $pre(\mathbf{w}) = w_2$ and $pre(\mathbf{b}) = b_2$.

The action model is displayed in figure 2.6. The action model consists of three actions. The precondition of an action corresponds to the answer of player 2. For instance, the action **r** models the action that player 2 answers "I am not holding the red card". Every action is accessible from every action by player

3. Player 3 is ignorant about which actual action is taken by player 2. The execution of the action model in a Kripke model results in a Kripke model consisting of 12 states.

We conclude by stating that definite knowledge and its development in the game Q-A is elegantly modeled by dynamic epistemic logic. The knowledge actions and initial knowledge distribution are separately modeled by respectively action models and a Kripke model. Not only simple knowledge actions in games of imperfect information can be modeled by action models, but also more complex knowledge actions. The knowledge development during the game is calculated by the execution of the action models in the Kripke model.

2.3.3 Strategic Knowledge

In the game Q-A, the players develop strategic knowledge if the players have knowledge about the used strategies. In chapter 2 we have given several examples of how the players of the game Q-A can develop strategic knowledge. In this subsection we try to model the strategic knowledge the players developed in these examples.

Always ask your own card Player 1 always asks about her own card. If it is common knowledge player 1 uses this strategy, the players commonly know which card player 1 is holding after the question of player 1, namely the card player 1 asks about. After the question of player 1, player 2 answers the question and the players further develop their knowledge about the distribution of the cards.

The question of player 1 alters the knowledge of the players, because the players have knowledge about the used strategy by player 1. We model the question by the action model $q_r = (A, R, pre)$:

- $A = \{\mathbf{q_r}\}$;
- $R(1) = R(2) = \emptyset$;
- $pre(\mathbf{q_r}) = r_1$.

The models of the answers of player 2 are unchanged. The answers "Yes" and "No" are respectively modeled by the action models yes_r and no_r. Figure 2.7 displays the knowledge development if the players commonly know the player 1 always asks about her own card.

Directly after the question, the only two possible deals of cards are rwb and rbw. Player 1 is holding the red and cannot distinguish the two deals of cards. Player 2 is holding a different card in the two deals of cards. Directly after the question of player 1, player 2 can distinguish every possible deal of cards and he knows the distribution of cards. Player 1 knows that player 2 knows the distribution of the cards.

From the answer of player 2, the players do not learn anything, because both players already know player 2 is not holding the red card. It is not possible for player 2 to answer "Yes", the action model yes_r results in an empty set of possible states.

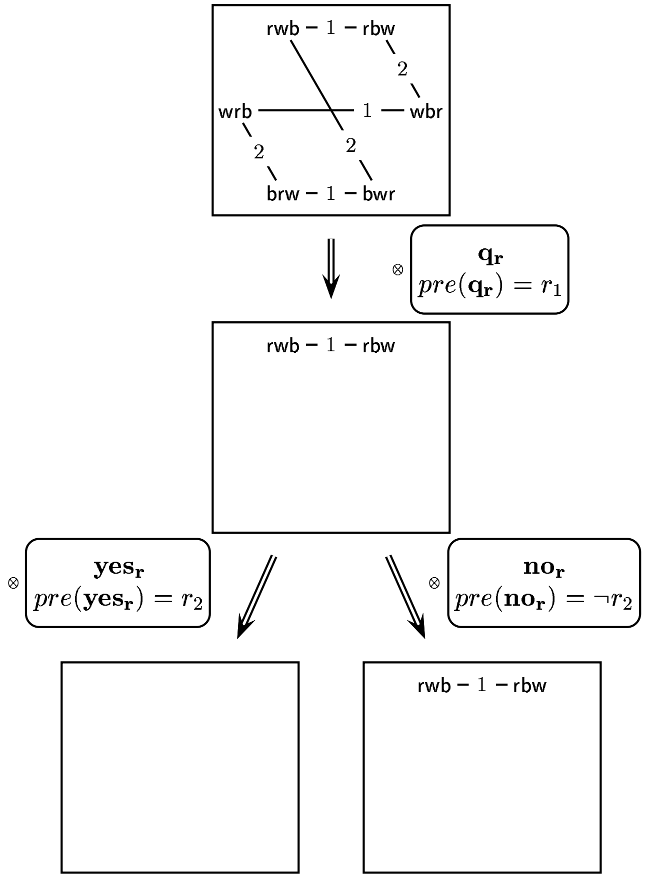

Figure 2.7: Always asking your own card

Strict Domination In chapter 2 we showed it is strictly dominated for player 1 to ask about the card she is holding herself. If player 1 is rational, she will never ask about her own card. Suppose it is common knowledge player 1 is rational. In this case the knowledge the players develop is also called rational knowledge. Player 1 asks about the red card. From this question learns the players that player 1 is not holding the red card. The question is modeled by the action model $q_r = (A, R, pre)$:

- $A = \{\mathbf{q_r}\}$;
- $R(1) = R(2) = \emptyset$;
- $pre(\mathbf{q_r}) = \neg r_1$.

Note that the precondition of the action $\mathbf{q_r}$ the opposite of the precondition of the action if it is common knowledge player 1 always asks about her own card. Figure 2.8 displays the knowledge development of the players if it is common knowledge player 1 is rational.

Directly after the question of player 1, both players do not hold the deals rwb and rbw as possible. It has become common knowledge player 1 is not holding

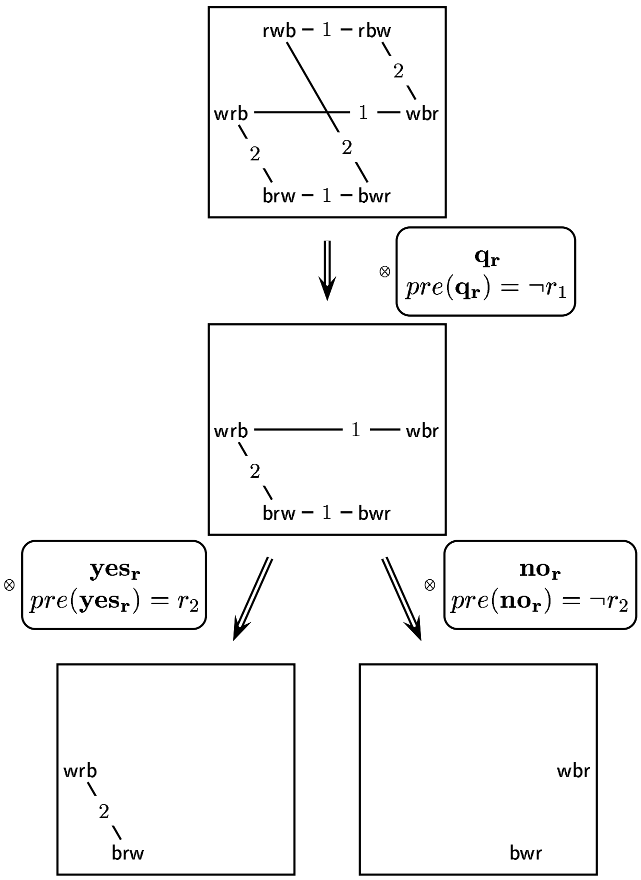

Figure 2.8: Strict Domination

the red card. If player 2 is not holding the red card, deals wbr and bwr, he knows which card is lying on the table. If player 2 is holding the red card, he does not learn anything from the question, because he already knew player 1 is not holding the red card. After the question, player 1 does not know which card is lying on the table, and whether player 2 knows which card is lying on the table.

It is a rational act of player 1 not to ask about the card she is holding. Suppose it is common knowledge player 1 is rational. Player 1 asks her question about the red card to learn something about the card on the table. If player 2 is not holding the red card, player 2 learns what card is lying on the table instead of player 1. After the question player 2 answers the question and also player 1 learns which card is lying on the table.

Common Knowledge If it is common knowledge player 1 uses a certain strategy, the strategic knowledge development of the players if player 1 asks about the red card is determined by the precondition of the action $\mathbf{q_r}$, modeling the question of player 1. For every example of strategic knowledge development in the game Q-A provided by chapter 1, we determine the precondition of the action $\mathbf{q_r}$.

Let it be common knowledge player 1 never asks about her own card. Her strategy is such that player 1 asks about the blue card when holding the white card and otherwise asks about the white card. The precondition of the action $\mathbf{q_r}$ is $\neg\top$, because player 1 will never ask about the red card.

In another example player 1 always asks the red card, independent of the card she is holding. If player 1 asks about the red card, the players do not learn anything about the card player 1 is holding, as she always asks about the red card. The precondition of the action is $\mathbf{q_r}$ is $r_1 \vee w_1 \vee b_1 = \top$. The action does not alter the knowledge of the players. The players only develop definite knowledge if player 1 always asks about the red card.

Suppose it is common knowledge the game Q-A is played according to the Nash-equilibrium described in chapter 2. Player 1 randomly asks about a card she is not holding. The strategic knowledge the players develop is the same as when it is common knowledge player 1 is rational. The precondition of the action $\mathbf{q_r}$ is $\neg r_1$.

Uncommon Knowledge We have given one example in which player 2 privately knew what strategy player 1 uses. Player 1 is not aware of the fact player 2 knows this. It is not common knowledge what strategy player 1 uses. Player 1 asks about the white card when holding the blue card and otherwise asks about the red card.

Suppose player 2 is holding the red card. After the question "Are you holding the red card ?", player 2 knows that player 1 must be holding the white or red card. If player 2 is holding the red card himself, player 1 must be holding the white card. Player 1 is ignorant about this knowledge of player 2 and has the incorrect knowledge that player 2 does not know which card player 1 is holding. The Kripke models we presented in this chapter cannot model the incorrect knowledge of player 1, because known facts are assumed to be true. In dynamic epistemic logic we cannot analyze the example of uncommon knowledge.

If it is common knowledge which strategy player 1 uses, the question of player 1 about the red card is modeled by the action $\mathbf{q_r}$. The precondition of the action is the interpretation of the question. Depending on the strategy of player 1, the action means player 1 is holding the red card, is not holding the red or does not mean anything. As a matter of fact, the question of player 1 can be interpreted in any way, depending on the strategy of player 1.

Given the strategy of player 1, her question is interpreted. The interpretation is done by hand. We find it hard to determine what the question means, because the question can have so many different meanings. The modeling of question is not trivial. For a satisfying model of the strategic knowledge development in the game Q-A and games of imperfect information in general, we find it necessary to be able to automatically interpret actions taken in a game, given the knowledge of the players about the used strategy. We find the analysis of the strategic knowledge by dynamic epistemic logic is unsatisfying, because the actions taken by the players need to be interpreted by hand, while the actions can have many different meanings.

2.3.4 Conclusions of a Card Game

In this section we analyzed the card game Q-A with dynamic epistemic logic. The analysis is concerned with the definite and strategic knowledge players can develop during the game. The definite knowledge players develop is elegantly modeled by dynamic epistemic logic.

Strategic knowledge cannot be satisfyingly modeled by dynamic epistemic logic. Action models modeling the strategic knowledge actions are too hard and not trivial to be determined by hand. Furthermore, in dynamic epistemic logic, the knowledge players develop if a player privately knows which strategies are used cannot be modeled.

2.4 Conclusions

In this chapter we have given an introduction to dynamic epistemic logic and the application of dynamic epistemic logic in games of imperfect information. Definite knowledge development in games of imperfect information is elegantly modeled by dynamic epistemic logic. We find an analysis of strategic knowledge development in games of imperfect information unsatisfying, because the determination of the action models is too hard.

At the beginning of this chapter our goals were to model definite knowledge and strategic knowledge. Definite knowledge can be elegantly modeled by dynamic epistemic logic. The modeling of strategic knowledge in dynamic epistemic logic is unsatisfying. Our goal is to come up with a model for strategic knowledge such that given the knowledge of the players about the used strategies, the knowledge development of the players can be calculated automatically, instead of determined by hand.

Chapter 3

Dynamic Epistemic Games

3.1 Introduction

The goal for this master thesis is to model definite and strategic knowledge and its development in games of imperfect information. In chapter 1, we described what the game knowledge of the players of a game of imperfect information is. In chapter 2 we modeled definite knowledge by dynamic epistemic logic. Furthermore we concluded that strategic knowledge development cannot be satisfyingly be modeled by dynamic epistemic logic.

When modeling strategic knowledge development, every action taken in a game needs to be interpreted as knowledge action and modeled by an action model. When modeling actions taken in a game, it is hard to tell how the action alters the knowledge of the players, given the strategies of the players. In other words, it is hard to transform an action taken in a game into an action model, given the strategies of the players, while the transformation is done by hand.

In other words, it is hard to tell how strategies influence knowledge, because it is hard to connect knowledge about the state of the game and to the knowledge about the used strategies. Up to now, the connection is done by hand, when transforming taken actions into action models. And precisely this transformation is hard to do. An analysis in dynamic epistemic logic has a high probability of being erroneous, due to human mistakes. To solve our problem we need to formally connect Kripke models to strategies. Kripke models and strategies need to be classed under one model of a game of imperfect information.

In this chapter we introduce the main contribution of our research, dynamic epistemic games. A dynamic epistemic game is a new model for games of imperfect information. In dynamic epistemic games, Kripke models and strategies are connected. In dynamic epistemic games, games of imperfect information are described in dynamic epistemic logic. The Kripke models and action models construct the game of imperfect information itself.

The dynamic epistemic game models the definite knowledge development of the players directly. Furthermore, the strategies of the players can be determined on the basis of a dynamic epistemic game. In one model, a dynamic epistemic game, the knowledge and strategies of the players are determined. To model the strategic players develop in a game of imperfect information, the Kripke models and strategies of the players are connected to calculate the Kripke models modeling the strategic knowledge of the players.

In section 3.2 we introduce dynamic epistemic games. In section 3.3 we show how definite and strategic knowledge is modeled. Furthermore we describe what the game knowledge of the players of a dynamic epistemic game is. Section 3.4 is an analysis of the game Q-A on strategies and definite and strategic knowledge development.

3.2 Dynamic Epistemic Games

What we will do in this chapter is describing games of imperfect information in dynamic epistemic logic. Dynamic epistemic logic provides the basic building blocks from which we will define a dynamic epistemic game. First we repeat the definitions of dynamic epistemic logic, then we introduce dynamic epistemic games.

3.2.1 Dynamic Epistemic Logic

Dynamic epistemic logic provides the building blocks for a dynamic epistemic game. A dynamic epistemic game is a collection of a Kripke models and sequences of action models. Kripke models and action models are the structures of which a dynamic epistemic game is build of. We introduce again Kripke and action models and other epistemic concepts in this subsection. Furthermore we comment specifically how the concepts will be used in the dynamic epistemic games. Provided with the building blocks, we present the dynamic epistemic game.

Definition 15 (Language)
Let the finite set of players N and the set of propositional atoms $\mathcal{P}$ be given. The *language* $\mathcal{L}_{\mathcal{P},N}$ is the smallest closed set for which holds:

- $p \in \mathcal{P} \Rightarrow p \in \mathcal{L}_{\mathcal{P},N}$;
- $\phi, \psi \in \mathcal{L}_{\mathcal{P},N} \Rightarrow \neg\phi, (\phi \wedge \psi) \in \mathcal{L}_{\mathcal{P},N}$;
- $\phi \in \mathcal{L}_{\mathcal{P},N}$ and $n \in N \Rightarrow K_n\phi \in \mathcal{L}_{\mathcal{P},N}$.

The propositional atoms $p \in \mathcal{P}$ describe relevant information to the players.

Definition 16 (Kripke Model)
Let a finite set of agents N and a set of propositional atoms $\mathcal{P}$ be given. A *Kripke model* M is a tuple (W, R, V):

- The set W is a nonempty set of states $\{w_1, \dots, w_{|W|}\}$;
- The *accessibility function* $R : N \rightarrow 2^{W \times W}$ is a function which assigns to each agent a set of ordered pairs of states. For every world w, w', w'' and agent n holds that: $(w, w) \in R(n)$ (reflexivity). If $(w, w') \in R(n)$ then $(w', w) \in R(n)$ (symmetry). If $(w, w'), (w', w'') \in R(n)$ then $(w, w'') \in R(n)$(transitivity);
- The *valuation function* $V : W \rightarrow 2^{\mathcal{P}}$ assigns to each state a set of propositional atoms.

A *Kripke world* is a pair (M, w) such that $w \in W$.

The states of the Kripke model will be viewed as states of the game. The accessibility relations determine which states of the game are indistinguishable to the players. The accessibility relation can be used for the same purpose as the information sets of a player in a game of imperfect information. The information sets also determined for a player which states of the game are indistinguishable for a player. A player has to take the same action in game states which are in the same information set. Given a Kripke model, a player has to take the same action in the states of the Kripke model which are indistinguishable for the player, i.e. the states which are accessible from each other for the player.

Definition 17 (Semantics for $\mathcal{L}_{\mathcal{P},N}$)
Let a Kripke model $M = (W, R, V)$ and the epistemic language $\mathcal{L}_{\mathcal{P},N}$ be given.

$$
\begin{array}{lcl}
M, w \models p & \Leftrightarrow & p \in V(w); \\
M, w \models \neg\phi & \Leftrightarrow & M, w \not\models \phi; \\
M, w \models \phi \wedge \psi & \Leftrightarrow & M, w \models \phi \text{ and } M, w \models \psi; \\
M, w \models K_n\phi & \Leftrightarrow & \text{For all worlds } w' \text{ such that } (w, w') \in R(n) \text{ holds: } M, w' \models \phi.
\end{array}
$$

The semantics for epistemic language are unchanged. Given a state, a player knows a sentence ϕ holds if and only if sentence ϕ holds in every accessible state. In a dynamic epistemic game, the states of the Kripke model are states of the game. We will use the semantics to determine which sentences hold in which states of the game. By doing this, we can determine what the knowledge of the players is states of the game.

Actions Actions taken in a game become dynamic epistemic actions. An action itself will model how the action alters the knowledge of the players.

Definition 18 (Action Model)
Let a finite set of agents N and the epistemic language $\mathcal{L}_{\mathcal{P},N}$ be given. An *action model* μ is a tuple (A, R, pre):

- The set A is a nonempty set of actions $\{a_1, \ldots, a_{|A|}\}$;
- The *accessibility function* $R : N \to 2^{A \times A}$ is a function which assigns to each agent a set of ordered pairs of actions. For every action a, a', a'' and agent n holds that: $(a, a) \in R(n)$ (reflexivity). If $(a, a') \in R(n)$ then $(a', a) \in R(n)$ (symmetry). If $(a, a'), (a', a'') \in R(n)$ then $(a, a'') \in R(n)$(transitivity);
- The *precondition function* $pre : A \to \mathcal{L}_{\mathcal{P},N}$ assigns to every action a precondition.

A *action event* is a pair (μ, a) such that $a \in A$.

We model actions taken in a game as action events. By doing this, we can keep track of knowledge development of the players during the game as consequences of the taken actions in the game. For instance, if a player answers he is not holding a specific card, the knowledge of the players alters. We model the answer as an action event, such that the action event models the knowledge development induced by the answer. The knowledge development is calculated by $\circ$-execution.

Definition 19 (∘-Execution)
Let a Kripke model $M = (W, R, V)$ and an action model $\mu = (A, R, pre)$ be given. The ∘-execution of action model μ in Kripke model M results in $M \circ \mu = (W', R', V')$ such that:

- The set of states is $W' = \{(w, a) \in W \times A | (M, w) \models pre(a)\}$;
- The accessibility function R' is such for every player n holds $((w, a), (w', a')) \in R'(n)$ iff $(w, w') \in R(n)$ and $(a, a') \in R(n)$.
- The valuation function V' is such that $V'(w, a) = V(w)$.

Overloading We overload the ∘ operator. The execution of an action event (μ, a) in a Kripke world (M, w) results in a Kripke world $(M, w) \circ (\mu, a) = (M \circ \mu, (w, a))$. The execution of a sequence of actions $(\mu^k)_{k=1...K}$ in Kripke model M results in Kripke model $M \circ (\mu^k)_{k=1...K} = ((M \circ \mu^1) \circ \ldots) \circ \mu^K$. The execution of the empty sequence $\varnothing$ in Kripke model M results in Kripke model $M \circ \varnothing = M$ The execution of a sequence of actions worlds $((\mu, a)^k)_{k=1...K}$ in Kripke world (M, w) results in Kripke world $(M, w) \circ ((\mu, a)^k)_{k=1...K} = (((M, w) \circ (\mu, a)^1) \circ \ldots) \circ (\mu, a)^K$.

The ∘ operator is a equivalent to the ⊗ operator with respect to action models and Kripke models. If we apply the ∘ operator to a Kripke model given an action model, the same Kripke model results as if we would have applied the ⊗ operator. We have chosen to introduce a new operator, because the ∘ operator is overloaded. An action event can be executed in a Kripke world. A sequence of action models can be executed in a Kripke model. And a sequence of action events can be executed in a Kripke world. All the execution are denoted by the same ∘-operator.

3.2.2 Dynamic Epistemic Game

Provided with the dynamic epistemic concepts introduced in the previous subsection, we define the dynamic epistemic game. A dynamic epistemic game is built of the dynamic epistemic structures as Kripke models and action models. The dynamic epistemic game serves as a basis on which we define a strategy of a player and the definite knowledge development of the players. Given a formal definition of the strategies of the players and a formal definition of the definite knowledge of the players, on the basis of one game model, we can connect strategies to knowledge and calculate the strategic knowledge of players within a game of imperfect information.

Definition 20 (Dynamic Epistemic Game)
Let a the language $\mathcal{L}_{\mathcal{P},N}$ be given. A *dynamic epistemic game* is a tuple (N, M, H, P).

- N is a finite set of *players*;
- M is a *Kripke model* (W, R, V);
- H is a finite set of *histories*. A *history* is a sequence of action models $(\mu)^k_{k=1...K}$. For the set of histories H the following properties hold:
    - The empty sequence $\varnothing$ is a member of H;

- ∘ Every subhistory of a history in H is also a member of H, i.e. if $(\mu)^k_{k=1\ldots K} \in H$ and $L < K$ then $(\mu)^k_{k=1\ldots L} \in H$.

Furthermore we define:

- ∘ A history $(\mu)^k_{k=1\ldots K} \in H$ is *terminal* iff there is no μ^{K+1} such that $(\mu)^k_{k=1\ldots K+1} \in H$ holds. The set of terminal histories is Z;
- ∘ An action model μ^{K+1} is *available* after history $(\mu)^k_{k=1\ldots K} \in H$, if $(\mu)^k_{k=1\ldots K+1} \in H$. The set of available action models after history h is $A(h)$.
- ∘ A *game sequence* is a sequence of actions events $\Lambda = ((\mu, a)^k)_{k=1\ldots K}$, such that $(\mu^k)_{k=1\ldots K} \in H$. The set of game sequences is Υ.

- The *player function* P assigns to each player a set of non-terminal histories. $P(n) \subseteq H$ is the set of histories in which player n decides which action is taken. P partitions the set of non-terminal histories, i.e. there exists no history such that $h \in P(n)$, $h \in P(m)$ and $n \neq m$.

  Given a player n, for every history $h \in P(n)$ holds:

  - ∘ For every action model μ, such that $\mu \in A(h)$ it holds that player n can distinguish all actions $a \in p_{1;3}(\mu)$. I.e. there exists no action $a' \in A$ such that $a \neq a'$ and $(a, a') \in p_{2;3}(\mu)(n)$;
  - ∘ For every action $a \in p_{1;3}(\mu)$, such that $\mu \in A(h)$ it holds that player n knows the precondition of action a holds or not in the world in which he takes an action. I.e. for every action $a \in p_{1;3}(\mu)$ and every state $w \in p_{1;3}(M \circ h)$ it holds that $(M \circ h, w) \models K_n pre(a) \vee K_n \neg pre(a)$.

The projection function $p_{i;j}(l)$ returns element i of the list l containing j number of elements.

The game does not consist of histories in which chance decides which action is taken. This is because of reasons of simplicity. In the above definition, no payoffs are assigned to terminal game states. Formally speaking, we have defined a game form, instead of a game.

The Kripke model M is the starting point of the game. The Kripke model models the initial definite knowledge of the players. From this starting point, several series of events are possible. Every possible series of events is a game sequence. The game sequence denotes which actions are actually taken within the game, and how these actions appear to the agents. The set of histories defines all possible game sequences and the appearance of these game sequences to the players.

State of the game The Kripke model M is the initial game model. One of the states of the Kripke model is the actual game state. The accessibility relations between the several possible states of the game, determine which states of the game are indistinguishable to the players. The game model and game state after respectively a history and game sequence is determined by $\circ$-execution.

Definition 21 (Game Model and Game State)
Let a game (N, M, H, P) be given. A *game model* is a Kripke model. The game model after history $h \in H$ is $M \circ h$. The *game state* after game sequence $\Lambda \in \Upsilon$ is $(M, w) \circ \Lambda$.

We call the resulting Kripke model the game model. The game state is the pointed equivalent of the game model. To be precise, the initial game model is the result of $M \circ \varnothing$. The resulting game models are always connected Kripke models. Every state is accessible from another state through a chain of accessibility relations.

Knowing which action to take In every state of a game model, a player initiates an action. At every game model, a set of action models is available The action models are available in the sense that the player can choose to take one of the actions of the available action models. The accessibility relation of the action model of the taken action determines how the action appears to the other players. So, the player initiates an action of an action model, the action is executed and has a certain appearance to the players. The game model is updated by the execution of the action model in the game model, and some player initiates another action, until a terminal game model is reached.

If a player wants to initiate an action in a Kripke world, the action needs to be executable in the Kripke world, i.e. the precondition of the action holds in the state of the game model. The player to initiate the action knows whether the precondition of the action holds in the Kripke world. If a player would be able to initiate an action of which he does not know whether the precondition holds, players could learn whether the precondition holds, by trying to initiate the action. For instance, a player could try to initiate an action, and if execution of the action does not succeed, the player learns the precondition does not hold and with this information initiate some other action. We do not allow these forms of learning.

A player knows which actions he takes and does not forget which actions he has taken. We assume the game is of *perfect recall.* To establish this, a player has to be able to distinguish actions he initiates. The available action models to player are such that the player can distinguish every action.

In a dynamic epistemic game, action models are assigned to game models. In every state of a game model, the player to move has the same actions available. By assigning action models to game models, and not to game states, the players commonly know which actions are available to the player to move in every state of a game model. If we would assign action models to game states, it would be possible to assign different sets of action models to states of the same game model. If a player then would initiate an action which is not available in all states of one game model, players learn about the state of the game, as they know which actions are available in which states of the game model. We also do not allow these forms of learning. Therefore it has to be common knowledge which action models are available to the player in a game model. We establish this by assigning the available action models to game models, and not to game states.

Knowing which player to take an action We assign the player to take an action to a game model. The player has to take an action in every state of the game model. It cannot be the case that different players take actions in states of the same game model. As a consequence, players commonly know who takes an action in the states of a game model. If this would not be the case, players would be able to learn about the state of the game by observing who takes the action. This is prohibited.

3.2.3 Strategies

A pure strategy of a player is a prescript that specifies the action chosen in each state of the game models in which the player has to take an action. First we define a Kripke strategy. A Kripke strategy is a function which assigns to each state of a game model an action. A strategy of a player n is a function which assigns to every history $h \in P(n)$ a Kripke strategy.

Definition 22 (Kripke Strategy)
Given a game model $M = (W, R, V)$, a set of available action models Γ and a player n, a *Kripke strategy* of player n is a function $t : W \rightarrow \{a \in p_{1;3}(\mu) | \mu \in \Gamma\}$ which assigns to each state w an action a. For every state $w \in W$ holds:

- If $t(w) = a$ then $(M, w) \models pre(a)$;
- if $(w, w') \in R(n)$ then $t(w) = t(w')$.

The set of Kripke strategies is denoted by T.

A Kripke strategy prescribes which action to take in the states of a game model, given the game model and a set of available action models. A Kripke strategy constrains the player in two ways. First of all, the precondition of the action taking in a state of the game model necessarily holds in the state. Secondly, a player takes the same action in states which are indistinguishable for the player.

Given the notion of a Kripke strategy we define a pure strategy of a player. A pure strategy is a function which assigns to every game model in which the player has to take an action a Kripke strategy. The Kripke strategy on its turn, prescribes which action is taken in every state of the game model.

Definition 23 (Pure Strategy)
Given a game (N, M, H, P), a *pure strategy* of player n is a function $s_n : H \rightarrow T$ which assigns to each history $h \in P(n)$ a Kripke strategy tk. The domain of the Kripke strategy is set of states of the game model corresponding to history h, i.e. $p_{1;3}(M \circ h)$. The range of the Kripke strategy is set of every action of the set of available action models after history h, i.e. $\{a \in p_{1;3}(\mu) | \mu \in A(h)\}$.

A *pure strategy profile* $\boldsymbol{s}$ is a list $(s_1, \ldots, s_{|N|})$ consisting of a pure strategy for each player. $\boldsymbol{s}_n$ is the strategy of player n according to the strategy profile. Let $h \in P(n)$, $\boldsymbol{s}(h) = \boldsymbol{s}_n(h)$.

A strategy profile completely prescribes which actions will be taken in the states of the game. We view a strategy profile as an function which assigns to every game state an action.

3.2.4 Example Hexas

As an example, we will model a game inspired by the game hexa as a dynamic epistemic game. We call the game Hexas. Van Ditmarsch presents in *Knowledge Games* [20] the game hexa:

> Consider the following game. There are three players. They are called 1, 2 and 3. There are three cards. The cards are called red, white and blue, or r, w, b (the colors of the Dutch flag). Every player is holding one card. Players can only see their own cards. A player can ask a question to another player. The question should always be answered. Also, after the question has been answered, the requesting player may announce that he knows what the deal of cards is. The first player to do so, wins the game. Players never lie, are prefect reasoners, know the kind of game they are player, etc. We call this game the hexa game.

For the game Hexas, we add the following. Player 1 begins by asking player 2 a question. Player 1 asks for one of two cards. Player 1 can ask for the red or white card, the red or blue card and the white or blue card. For instance player 1 asks: "Do you have the red or white card ?". Player 2 can give two responses. He publicly announces "I do not have neither of the cards." or he privately informs player 1 he is holding one of the two cards, by showing player 1 the card he is holding. Player 3 does not see what card player 2 is showing, but does learn player 2 shows his card to player 1. After player 2 has responded to the question, player 1 may announce that she knows the deal of cards. If she does not announce that she knows the deal of cards, for whatever reason, it is player 2's turn and player 2 asks for one of two cards, and so on.

Dynamic Epistemic Game We partly model the game as a dynamic epistemic game. A deal is denoted by a sequence of letters. rwb denotes player 1 is holding the red card, player 2 is holding the white card and player three is holding the blue card. Let c_n be the propositional atom expressing player n is holding card c. Hexas is modeled as as the dynamic epistemic game (N, M, H, P):

- $N = \{1, 2, 3\}$;
- $M = (W, R, V)$:
  - $W = \{\mathsf{rwb}, \mathsf{rbw}, \mathsf{wrb}, \mathsf{wbr}, \mathsf{brw}, \mathsf{bwr}\}$;
  - $R(1) = \{(\mathsf{rwb}, \mathsf{rbw}), (\mathsf{wrb}, \mathsf{wbr}), (\mathsf{brw}, \mathsf{bwr})\}$, $R(2) = \{(\mathsf{rwb}, \mathsf{bwr}), (\mathsf{wrb}, \mathsf{brw}), (\mathsf{rbw}, \mathsf{wbr})\}$ and $R(3) = \{(\mathsf{rwb}, \mathsf{wrb}), (\mathsf{rbw}, \mathsf{brw}), (\mathsf{wbr}, \mathsf{bwr})\}$;
  - $V(\mathsf{rwb}) = \{r_1, w_2, b_t\}$, $V(\mathsf{rbw}) = \{r_1, b_2, w_t\}$, $V(\mathsf{wrb}) = \{w_1, r_2, b_t\}$, $V(\mathsf{wbr}) = \{w_1, b_2, r_t\}$, $V(\mathsf{wbr}) = \{w_1, b_2, r_t\}$, $V(\mathsf{brw}) = \{b_1, r_2, w_t\}$ and $V(\mathsf{bwr}) = \{b_1, w_2, r_t\}$.
- $H = \{\varnothing, (q_{rw}), (q_{rb}), (q_{wb}), (q_{rw}, no_{rw}), (q_{rb}, no_{rb}), (q_{wb}, no_{wb}), (q_{rw}, show_{rw}), (q_{rb}, show_{rb}), (q_{wb}, show_{wb}), (q_{rw}, no_{rw}, win), (q_{rb}, no_{rb}, win), (q_{wb}, no_{wb}, win), (q_{rw}, show_{rw}, win), (q_{rb}, show_{rb}, win), (q_{wb}, show_{wb}, win), (q_{rw}, no_{rw}, nowin), (q_{rb}, no_{rb}, nowin), (q_{wb}, no_{wb}, nowin), (q_{rw}, show_{rw}, nowin), (q_{rb}, show_{rb}, nowin), (q_{wb}, show_{wb}, nowin)\}$;
  - $q_{rw} = (\{\mathbf{q_{rw}}\}, \{(1, \emptyset)\}), (2, \emptyset)\}), (3, \emptyset)\})\}, \{(\mathbf{q_{rw}}, \mathsf{T})\})$;
  - $no_{rw} = (\{\mathbf{no_{rw}}\}\ , \{(1, \emptyset)\}), (2, \emptyset)\}), (3, \emptyset)\}, \{(\mathbf{no_{rw}}, \neg r_2 \wedge \neg w_2)\})$;
  - $show_{rw} = (\{\mathbf{r}, \mathbf{w}\}, \{(1, \emptyset), (2, \emptyset), (3, \{(\mathbf{r}, \mathbf{w})\})\}, \{(\mathbf{r}, r_2), (\mathbf{w}, w_2)\})$;
  - $win = (\{\mathbf{win}\}, \{(1, \emptyset), (2, \emptyset), (3, \emptyset)\}, \{(\mathbf{win}, \mathsf{win})\})$;

- $\circ$ $nowin = (\{\mathbf{nowin}\}, \{(1,\emptyset), (2,\emptyset), (3,\emptyset), \{(\mathbf{nowin}, \mathsf{T})\});$

- $P(1) = \{\varnothing, (q_{rw}, no_{rw}), (q_{rb}, no_{rb}), (q_{wb}, no_{wb}), (q_{rw}, show_{rw}), (q_{rb}, show_{rb}), (q_{wb}, show_{wb})\}$, $P(2) = \{(q_{rw}), (q_{rb}), (q_{wb}), (q_{rw}, no_{rw}, nowin), (q_{rb}, no_{rb}, nowin), (q_{wb}, no_{wb}, nowin), (q_{rw}, show_{rw}, nowin), (q_{rb}, show_{rb}, nowin), (q_{wb}, show_{wb}, nowin)\}$ and $P(3) = \emptyset$.

$\mathsf{win} = (K_1 r_1 \vee K_1 w_1 \vee K_1 b_1) \wedge (K_1 r_2 \vee K_1 w_2 \vee K_1 b_2) \wedge (K_1 r_3 \vee K_1 w_3 \vee K_1 b_3).$

We worked out five action models. The question "Do you have the red or white card ?" is modeled by the action model q_{rw}. The other questions are modeled by the equivalent action models q_{rb} and q_{wb}. Player 2 can give two responses, either he does not have one of the cards asked about, or he shows one of the cards to player 1. The two responses are respectively modeled by the action models no_{rw} and $show_{rw}$. After the response of player 2, player 1 may announce she has won. Announcing to have won corresponds to action model win. Not announcing to have won, is modeled by action model $nowin$. The game is graphicly displayed in figure 3.1.

Strategies We give an example of a strategy for player 1 and player 2. Let s_1 be a strategy for player 1.

$$s_1 = \begin{cases} s_1(\varnothing)(\mathsf{brw}) = s_1(\varnothing)(\mathsf{bwr}) = \mathbf{q_{rw}} \\ s_1(\varnothing)(\mathsf{rwb}) = s_1(\varnothing)(\mathsf{rbw}) = \mathbf{q_{wb}} \\ s_1(\varnothing)(\mathsf{wrb}) = s_1(\varnothing)(\mathsf{wbr}) = \mathbf{q_{rw}} \\ \\ s_1(q_{rw}, no_{rw})(\mathsf{rbw}, \mathbf{q_{rw}}, \mathbf{no_{rw}}) = \mathbf{win} \\ s_1(q_{rw}, no_{rw})(\mathsf{wbr}, \mathbf{q_{rw}}, \mathbf{no_{rw}}) = \mathbf{nowin} \\ \\ s_1(q_{rw}, show_{rw})(\mathsf{rwb}, \mathbf{q_{rw}}, \mathbf{show_{rw}}) = \mathbf{win} \\ s_1(q_{rw}, show_{rw})(\mathsf{wrb}, \mathbf{q_{rw}}, \mathbf{show_{rw}}) = \mathbf{nowin} \\ s_1(q_{rw}, show_{rw})(\mathsf{brw}, \mathbf{q_{rw}}, \mathbf{show_{rw}}) = \mathbf{win} \\ s_1(q_{rw}, show_{rw})(\mathsf{bwr}, \mathbf{q_{rw}}, \mathbf{show_{rw}}) = \mathbf{win} \end{cases}$$

s_2 is a strategy for player 2.

$$s_2 = \begin{cases} s_2(q_{rw})(\mathsf{rwb}, \mathbf{q_{rw}}) = s_2(q_{rw})(\mathsf{bwr}, \mathbf{q_{rw}}) = \mathbf{w} \\ s_2(q_{rw})(\mathsf{wrb}, \mathbf{q_{rw}}) = s_2(q_{rw})(\mathsf{brw}, \mathbf{q_{rw}}) = \mathbf{r} \\ s_2(q_{rw})(\mathsf{rbw}, \mathbf{q_{rw}}) = s_2(q_{rw})(\mathsf{wbr}, \mathbf{q_{rw}}) = \mathbf{no_{rw}} \end{cases}$$

Modeling the game We have given a partial model of the game we described. The set of histories is incomplete. To complete the game model, we should add action models after the histories (q_{rb}), (q_{wb}), $(q_{rw}, no_{rw}, nowin)$ and $(q_{rw}, show_{rw}, nowin)$. What we have done is given a dynamic epistemic game model of the question "Do you have the red or white card ?", the response of player 2 to this question and announcement of player 1 after the response of player 2. We have modeled the questions of player 1, the responses of player 2 and the announcements of player 1 separately as action models. The action models are combined with a Kripke model which together construct the dynamic epistemic game. The modeling itself is modular, as we can model the several parts of the game as individual objects.

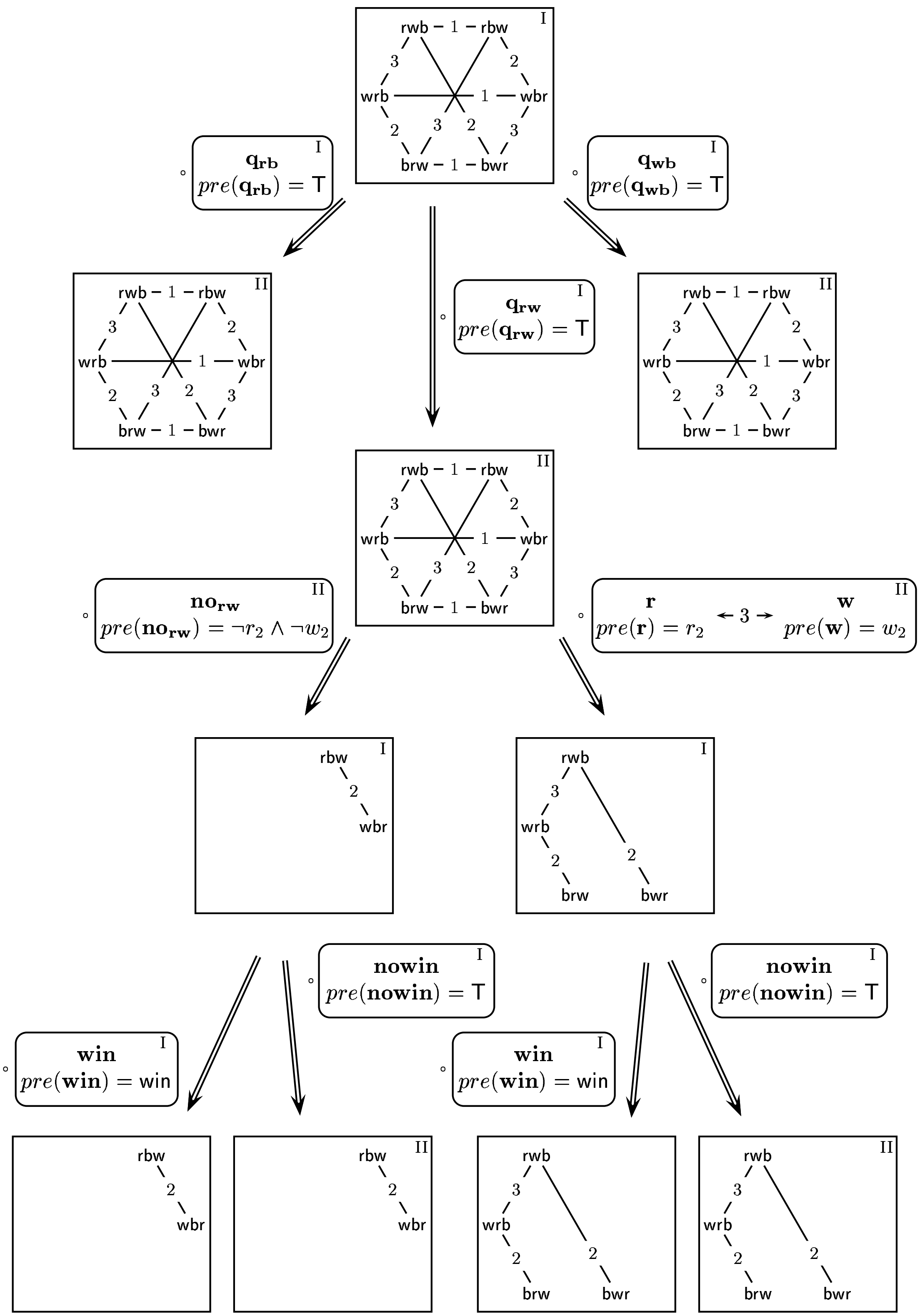

Figure 3.1: The game Hexas

Given the dynamic epistemic game, the game models, displayed in figure 3.1, are calculated by $\circ$-execution. The game models are models of the definite knowledge development of the players. By modeling the game as a dynamic epistemic game, the definite knowledge development of the players is modeled.

Modeling actions The question and responses are modeled separately, because the actions are initiated by different players. The questions player 1 can ask player 2 are modeled by the action models q_{rb}, q_{rw} and q_{wb}. The game models after the questions are identical to the initial game model, as the question itself does not alter the knowledge of the players. The game model is unchanged, because the precondition of the question is T, the question can always be asked, independent of the card player 1 holding.

The knowledge development induced by the question of player 1 is modeled by the action models modeling the responses of player 2 to the question. Player 2 can respond in two ways. Either he announces he is not holding one of the two cards, or he shows one of the two cards to player 1. The announcement that he is not holding one of the two cards is modeled by the action model no_{rw}. Showing one of the two cards is modeled by the action model $show_{rw}$. The action models no_{rw} and $show_{rw}$ together consist of 3 actions, $\mathbf{no_{rw}}$, **w** and **r**. These actions are the available actions to player 2 after the question of player 1. The action player 2 initiates is determined by the card player 2 is holding. If player 2 is holding the red card, he will take action **r**. If player is holding the white card, he will take action **w**. If player 2 is not holding the red or white card, he will take action $\mathbf{no_{rw}}$. Player 2's strategy is determined by the card he is holding, because in each case, the precondition of only one action is consistent with the card he is holding.

The game model resulting from the execution of action model $show_{rw}$ consists of the states rwb, wrb, brw and bwr. In the states rwb and bwr, the action **w** was executed. The states wrb and brw are the result of the execution of action **r**. Before the actions were taken, player 1 could not distinguish brw from bwr. In both states player 1 holding the blue card. After the execution of the action model $show_{rw}$ player 1 can distinguish the two states. Player 1 can distinguish the two states, because different actions were executed in the two states. Up to this moment, knowledge development of players was always modeled as a deletion of states in a Kripke model. In this example player 1 learns which card player 2 is holding. This knowledge development is not modeled by the deletion of one or more states, but by the deletion of an accessibility relation.

Player 1 may announce that she has won, after the response of player 2. The announcement implies player 1 knows the deal of cards. This implication corresponds to the sentence win. The sentence states player 1 knows which card player 1 is holding, which card player 2 is holding and player 1 knows which card player 3 is holding. The announcement is modeled by the action model *win*, which consists of the action **win**. The precondition of the action is the sentence win. From the announcement by player 1 to have won, players do not learn anything about the deal of cards. The game model is unchanged, as the players already knew player 1 knows the deal of cards. In every state of the game models before the action model *win*, player 1 can distinguish every deal of cards, so player 1 knows the deal of cards. After the action **win** the game stops.

Player 1 can also choose to let the game continue, by taking action **nowin**. In this case, player 1 does not declare to have won the game, even though she knows the deal of cards. We do not obligate player 1 to win the game, if she knows the deal of cards. Van Ditmarsch states in *Knowledge Games* [20] a player cannot take an action such as **nowin** if the player knows the deal of cards. In Van Ditmarsch' case a player declares not to be able to win the game by announcing not to have won.

Strategies A strategy of a player prescribes the player which action to take in every state of the game model in which the player has to take an action. A strategy of player 1 consists of two parts. First of all, the strategy prescribes her which question to ask, depending on the deal of cards. Secondly, the strategy determines to announce to have won or not, depending on the response of player 2 and the deal of cards. In this example, player 1 chooses to ask a question such that, she asks about the red and white card when holding the blue or white card and asks about the white or blue card when holding the red card. Player 1 is inconstant about declaring to have won or not. Only if player 1 is not holding the white card, she announces to have won after the response of player 2. The strategy is such that player 1 does not choose to win, when holding the white card. This is a strategy of player 1. We do not know whether this is a good strategy.

The strategy of player 1 specifies which action to take in the states rwb and rbw after the response of player 2. Player 1 knows these states will not be reached as player 1 does not ask about the red and white card, when holding the red card. Nevertheless, the strategy of player 1 species which action to take in these states.

We have worked out the pure strategy of player 2 for the case player 1 takes action $\mathbf{q_{rw}}$. In every state, the precondition of only one action holds, so the pure strategy is completely determined by the precondition of his available actions. Player 2 has to show the white card, when holding the white card and has to show the red card when holding the red card. If player 2 is not holding either the white or red card, he has to announce is not holding the red or white card.

Conclusions Hexas We have modeled the game as a dynamic epistemic game. For every player we have given a pure strategy. Furthermore we have commented on the modeling of the game, the actions and the strategies of the players.

3.2.5 Comparison with Game Theory

A game of imperfect information can be modeled as the game model provided by game theory and as a dynamic epistemic game. We give a comparison of a game of imperfect information and a dynamic epistemic game.

A game of imperfect information an be viewed as a tree. The nodes of the tree are the possible states of the game. The arrows between the nodes are actions. Every non-terminal node is labeled with the player to take an action at that state of the game. For every player, the set of nodes in which the player has to take an action is partitioned into information sets. Players cannot distinguish states of the games, which are in the same information set.

In this chapter we introduced a dynamic epistemic game. A dynamic epistemic game can also be viewed as a tree. The nodes of the tree are game models, Kripke models modeling the definite knowledge of the players. The arrows between the game models are labeled with action models. A game model is calculated by the execution of the action model in the preceding game model. Every non-terminal game model is assigned to a player, the player to take an action in every state of the game model.

Comparison When modeling a game as a dynamic epistemic game, one models the initial knowledge of the players by a Kripke model. All the actions in the game are modeled by action models. The Kripke model, together with a set of sequences of action models constructs a dynamic epistemic game. The several aspects of the game can be modeled separately and then put together. A dynamic epistemic game is modular, so to speak.

For a dynamic epistemic game, the accessibility relations of the Kripke model and the action models can be determined separately, as a dynamic epistemic game has this modular property. The accessibility relations of the game models are calculated by execution of the action models. When modeling a game of imperfect information, one has to keep track of the whole game tree, to determine the information sets. The modeling of a dynamic epistemic game is easier, as one can concentrate on the individual objects when constructing a dynamic epistemic game.

The modeler of a game of imperfect information needs to determine the information sets for the whole tree. The information sets and the accessibility relations determine the indistinguishable of players over the states of the game. For a dynamic epistemic game, the accessibility relations for the action models and Kripke model are determined separately, because a dynamic epistemic game is modular. A whole game-tree together with information sets are hard to interpreted, as one cannot focus on the different aspects of the game. This is a consequence of the lack of surveyability of the information sets. A dynamic epistemic game gives insight to a game, because of the modular property of dynamic epistemic games. Therefore we find dynamic epistemic games more elegant.

Constraining games A dynamic epistemic game is constrained in more ways than the game model provided by game theory. The constraints determine which games we can and which games we cannot model. Here we give an overview of how the dynamic epistemic game is constrained.

1. The accessibility relations of the game models and action models are reflexive, symmetrical and transitive. In these models, the knowledge of the players are facts. The players do not know something which is incorrect. Furthermore, the players know what they do know and what they do not know. We cannot model games in which players use incorrect knowledge or in which players do not know what they know and what they do not know. So cheating and deception cannot be modeled. This constraint also holds for a game of imperfect information.

2. We use a Kripke model as a game model. The Kripke models are the result of the execution of a sequence of action models. Every state is the result of the same sequence of action models. In every state, it is common knowledge what the history of action models is. The players commonly

know which action model has been executed and who has taken an action. Games in which players are not informed whether a player has taken an action or not cannot be modeled by dynamic epistemic games. *Selten's horse* (Binmore, *Fun & Games*, page 538 [2]) is a game in which player 3 does not know whether player 2 has taken an action or not. This game cannot be modeled as a dynamic epistemic game. This constraint does not hold for a game of imperfect information.

3. The available actions are assigned to game models. In every state of a game model, the player to take an action has the same actions available. It is common knowledge which actions are available. This constraint does not hold for a game of imperfect information. Again, Selten's horse is a game in which it is not common knowledge which actions are available to the players. Player 3 does not know whether player 2 can take an action or not, and so she does not know which actions are available to player 2. In a game of imperfect information, only the player to take an action needs to know the available actions.

4. The player function assigns game models to players. In every state of a game model, the same player takes an action. In a dynamic epistemic game, it is common knowledge who is to take an action. In a game of imperfect information, only the player to take an action needs to know that he will take an action. Selten's horse is a game in which the players do not commonly know which player is to take an action.

5. The player to move can distinguish all actions he can initiate. In a dynamic epistemic game, players have perfect recall. The players remember which actions they have taken. In a game of imperfect information, most of the games are assumed to be of perfect recall, but games of imperfect information are not necessarily of perfect recall. In those games, the players are allowed to forget which action they took.

In dynamic epistemic games strategies and definite knowledge are modeled by the same model. This is done by modeling a game with dynamic epistemic building blocks. But these building blocks also constraints the games we can model as dynamic epistemic games. We have given five constraints a game has fulfill to be modeled as a dynamic epistemic game. These constraints do not always hold for the game model provided by game theory for a game of imperfect information. Some games can be modeled by game theory, but cannot be modeled as a dynamic epistemic game.

3.2.6 Conclusions Dynamic Epistemic Games

We have introduced a dynamic epistemic game. A dynamic epistemic game serves as a basis to define a strategy of a player in a game, and also models the definite knowledge development of the players during the game. We wanted to connect strategies to knowledge to model strategic knowledge. In this model, we can connect strategies to knowledge, as the game is the basis for strategies and knowledge. On the basis of this model, we will model the strategic knowledge of players and its development.

3.3 Knowledge

In a game, we can distinguish three types of knowledge players have. Players have game knowledge, definite knowledge and strategic knowledge. In this section, we explain how these forms of knowledge are implemented in dynamic epistemic games.

3.3.1 Game Knowledge

The players of a game have knowledge about the game they are playing. We call this game knowledge. Here we give an overview of the game knowledge of the players of a dynamic epistemic game. During the game the players commonly know the set of players, the constraints on a dynamic epistemic game, the constraints on a strategy of a player and all the possible game models, and how the game model can be reached through a sequence of action models. For every game model the player commonly know the game model they are in, the available actions to the player to take an action, the player to take an action. We have summarized the game knowledge of the players during a dynamic epistemic game.

3.3.2 Definite Knowledge

Definite knowledge is the knowledge players develop, because of explicit information exchange. For instance, in the game Hexas player 1 learns what the deal of cards is, when player 2 responds to the question of player 1. If player 1 is a perfect logician and player 2 never lies player 1 will always know the deal of cards after the response. Definite knowledge is objective, it will always be developed, under the assumption of perfect logicians and truth telling players. The definite knowledge players develop during the game, is inherent to the game, because of this objective property of definite knowledge. We can give for every game exactly one model which models the definite knowledge and its development.

A dynamic epistemic game uses the same structures, Kripke models and action models, as dynamic epistemic logic to model definite knowledge. The initial knowledge distribution is modeled by a Kripke model. Actions taken in the game are modeled by action models. The action models model how an action in a game alters the knowledge of the players. In this case we are only talking about definite knowledge. Definite knowledge and its development is modeled by a Kripke model, the set of sequences of action models and the $\circ$-execution of the sequences action models in the initial Kripke model. The result of the $\circ$-execution are the game models. A game model is a Kripke model which models the definite knowledge of the players.

A game model consists of a set of states, together with accessibility for every player and a valuation function. The states are states of the game. To every state a set of propositions is assigned by the valuation function. The propositions are about relevant information about the game. For instance, the propositions of the game Hexas are about who is holding which card. For every state of the game, we keep track of which propositions hold, and which do not.

Given such a valuation function for every state of the game, epistemic sentences can be tested in the states of a game model using the semantics of the epistemic languages. Here we give the recipe to determine the definite knowledge of the players in a game. As an example, we show player 3 knows player 1 knows which card player 3 is holding if the actual deal of cards is rwb after player 2 has shown his red or white card to player 1 in the game Hexas.

1. We model the game as a dynamic epistemic game. The initial knowledge distribution is modeled as a Kripke model. The possible actions taken in the game, are modeled by action models.

2. The game model to which the state of the game in which we are interested belongs is determined by $\circ$-execution. We execute the history corresponding to state of the game in the Kripke model. In our example, the history $(q_{rw}, show_{rw})$ is executed in the Kripke model, which gives us the game model. The game model is displayed in figure 3.1.

3. The sentence in natural language we want to test is translated to an epistemic sentence. "Player 3 knows player 1 knows which card player 3 is holding" corresponds to the epistemic sentence $K_3 K_1 b_3$.

4. The semantics of the epistemic language are applied. The semantics state a player knows a sentence holds in a state if the sentence holds in every state which is indistinguishable to the player. Our sentence holds in state rwb if player 1 knows that player 3 is holding the blue card in the states rwb and wrb. rwb and wrb are indistinguishable to player 3. Player 1 knows player 3 is holding the blue card, as in the states rwb and wrb the accessible states for player 1 are the states itself and in both states it holds that player 3 is holding the blue card.

Given this recipe, we can determine the definite knowledge and its development of the players in a game of imperfect information precisely.

3.3.3 Strategic Knowledge

Strategic knowledge is developed when players have knowledge over the used strategies. The players have some strategic information of the opponents, which is used to deduce knowledge about the state of the game. The strategic information can have several forms. For instance, a player may know his opponent will never take a certain action in a game state. If the player observes this action, he will deduce the state of the game is not the game state in which his opponent would never take that action. Another example of strategic information is the case in which the players commonly know all players are rational. In this case, the players know strictly dominated strategies will not be used. The knowledge rational players develop if it is common knowledge the players are rational is called rational knowledge. We will model the case in which the players commonly know which pure strategies are used. We give an example to clarify the case.

A game of Hexas is played. It has become common knowledge the players use strategies according to the strategy profile $(s_1, s_2, \emptyset)$. The strategy of player 1 is such that player 1 asks about the red and white card when holding the blue or white card and asks about the white and blue card when holding the red card, and it is common knowledge player 1 will do so. The cards are dealt over the players and the game begins. Player 1 asks player 2 about the red and white card. The players know player 1 will never ask about the red and white card if player 1 is holding the red card herself. The players deduce that the deals rwb and rbw are impossible. When modeling this knowledge development by Kripke models, it can be modeled as a deletion of possible states rwb and rbw.

When executing action models in dynamic epistemic games, the precondition of an action rules out states in which the precondition does not hold. If players commonly know the used strategies by the players, we view the strategies as a precondition. The execution of an action in a state results only in a new state if the precondition of the state holds, and the strategy profile prescribes that the action is taken in that state.

Definition 24 (•-Execution)
Let a Kripke model (W, R, V), an action model $\mu = (A, R, pre)$ and a Kripke strategy t be given. The •-execution of the action model μ in game model M results in $M \bullet (\mu, t) = (W', R', V')$ such that:

- The set of worlds is $W' = \{(w, a) \in W \times A | (M, w) \models pre(a) \text{ and } t(w) = a\}$;
- The accessibility function R' is such for every player n holds $((w, a), (w', a')) \in R'(n)$ iff $(w, w') \in R(n)$ and $(a, a') \in R(n)$.
- The valuation function V' is such that $V'(w, a) = V(w)$.

Overloading We overload the • operator. The execution of a pointed action model (μ, a) in a Kripke world (M, w) results in a Kripke world such that $(M, w) \bullet ((\mu, a), t) = (M \bullet (\mu, t), (w, a))$. Given a strategy profile $\boldsymbol{s}$, the execution of a sequence of actions $(\mu^k)_{k=1...K}$ in Kripke model M results in Kripke model $M \bullet ((\mu^k)_{k=1...K}, \boldsymbol{s}) = ((M \bullet (\mu^1, \boldsymbol{s}(\mu^1))) \bullet \ldots) \bullet (\mu^K, \boldsymbol{s}((\mu^k)_{k=1...K})$. The execution of a sequence of actions worlds $((\mu, a)^k)_{k=1...K}$ in Kripke world (M, w) results in Kripke world $(M, w) \bullet (((\mu, a)^k)_{k=1...K}, \boldsymbol{s}) = (M \bullet ((\mu^k)_{k=1...K}, \boldsymbol{s}), (w, a^1 \ldots a^K))$.

•-execution is an extension of ∘-execution. We added the strategy of a player as an extra precondition. If it is common knowledge which strategies the players use, the strategic knowledge of the players during the game can be calculated by •-execution.

Example Hexas The game Hexas is played and it is common knowledge the players use strategies according to strategy profile $\boldsymbol{s} = (s_1, s_2, \emptyset)$. Player 1 uses strategy s_1:

$$s_1 = \begin{cases} s_1(\varnothing)(\mathsf{brw}) = s_1(\varnothing)(\mathsf{bwr}) = \mathbf{q_{rw}} \\ s_1(\varnothing)(\mathsf{rwb}) = s_1(\varnothing)(\mathsf{rbw}) = \mathbf{q_{wb}} \\ s_1(\varnothing)(\mathsf{wrb}) = s_1(\varnothing)(\mathsf{wbr}) = \mathbf{q_{rw}} \\ \\ s_1(q_{rw}, no_{rw})(\mathsf{rbw}, \mathbf{q_{rw}}, \mathbf{no_{rw}}) = \mathbf{win} \\ s_1(q_{rw}, no_{rw})(\mathsf{wbr}, \mathbf{q_{rw}}, \mathbf{no_{rw}}) = \mathbf{nowin} \\ \\ s_1(q_{rw}, show_{rw})(\mathsf{rwb}, \mathbf{q_{rw}}, \mathbf{show_{rw}}) = \mathbf{win} \\ s_1(q_{rw}, show_{rw})(\mathsf{wrb}, \mathbf{q_{rw}}, \mathbf{show_{rw}}) = \mathbf{nowin} \\ s_1(q_{rw}, show_{rw})(\mathsf{brw}, \mathbf{q_{rw}}, \mathbf{show_{rw}}) = \mathbf{win} \\ s_1(q_{rw}, show_{rw})(\mathsf{bwr}, \mathbf{q_{rw}}, \mathbf{show_{rw}}) = \mathbf{win} \end{cases}$$

s_2 is the used strategy by player 2:

$$s_2 = \begin{cases} s_2(q_{rw})(\mathsf{rwb}, \mathbf{q_{rw}}) = s_2(q_{rw})(\mathsf{bwr}, \mathbf{q_{rw}}) = \mathbf{w} \\ s_2(q_{rw})(\mathsf{wrb}, \mathbf{q_{rw}}) = s_2(q_{rw})(\mathsf{brw}, \mathbf{q_{rw}}) = \mathbf{r} \\ s_2(q_{rw})(\mathsf{rbw}, \mathbf{q_{rw}}) = s_2(q_{rw})(\mathsf{wbr}, \mathbf{q_{rw}}) = \mathbf{no_{rw}} \end{cases}$$

The Kripke models modeling the strategic knowledge of the players during this game are calculated by •-execution. The Kripke models are displayed in figure 3.2. The figure is no representation of the game, it only displays the development of the strategic knowledge of the players during the game.

After the deal of the cards player 1 asks about the red and white card. The players deduce player 1 cannot be holding the red card. This knowledge development is modeled by the deletion of the states rwb and rbw. Then player 2 responds to the question. If player 2's card is blue, his response is $\mathbf{no_{rw}}$, it becomes common knowledge that wbr is the deal of cards, as only this state survives the action $\mathbf{no_{rw}}$.

If player 2 is holding the white card, player 2 has to take action **w** and it becomes common knowledge bwr is the deal of cards. In the case player 2 is holding the red card, player 2 takes action **r** and player 1 and 3 learn the deal of cards. Player 2 knows his opponents are informed about the deal of cards, but he can still not distinguish deal wrb from brw.

The actions **win** and **nowin** do inform the players about the deal of cards in contrast with the case in which the players are not informed about each other strategies. After the announcement about winning or not winning the game, player 2 can distinguish the deals wrb and brw, because player 2 knows player 1 announces to have won in state brw and announces in state wrb not to have won.

The game models in figure 3.1 modeling definite knowledge are connected models. Every state is reachable from another state through a chain of accessibility relations. Connectivity does not hold for the Kripke models modeling the strategic knowledge of the players. The •-execution of a connected action model in a connected Kripke model does not necessarily result in a connected Kripke model. So in a game it can become common knowledge what the state of the game is due to strategic knowledge, while the game models modeling the definite knowledge of the players suggests at least one player is ignorant about the precise state of the game.

By the application of the •-execution to a dynamic epistemic game, we can model the strategic knowledge if it is common knowledge which pure strategies are by the players.

3.4 Analysis of the Card Game

Q-A is a card game consisting of two players, a table and three cards. The cards are dealt over the players and the table. The goal of the players is to correctly guess the card lying on the table. During the game, the players develop knowledge, through information exchange. Also the players can develop knowledge about the deal of card if the players have knowledge about the used strategies. We are interested in the knowledge the players develop during the game, and how we can model this knowledge.

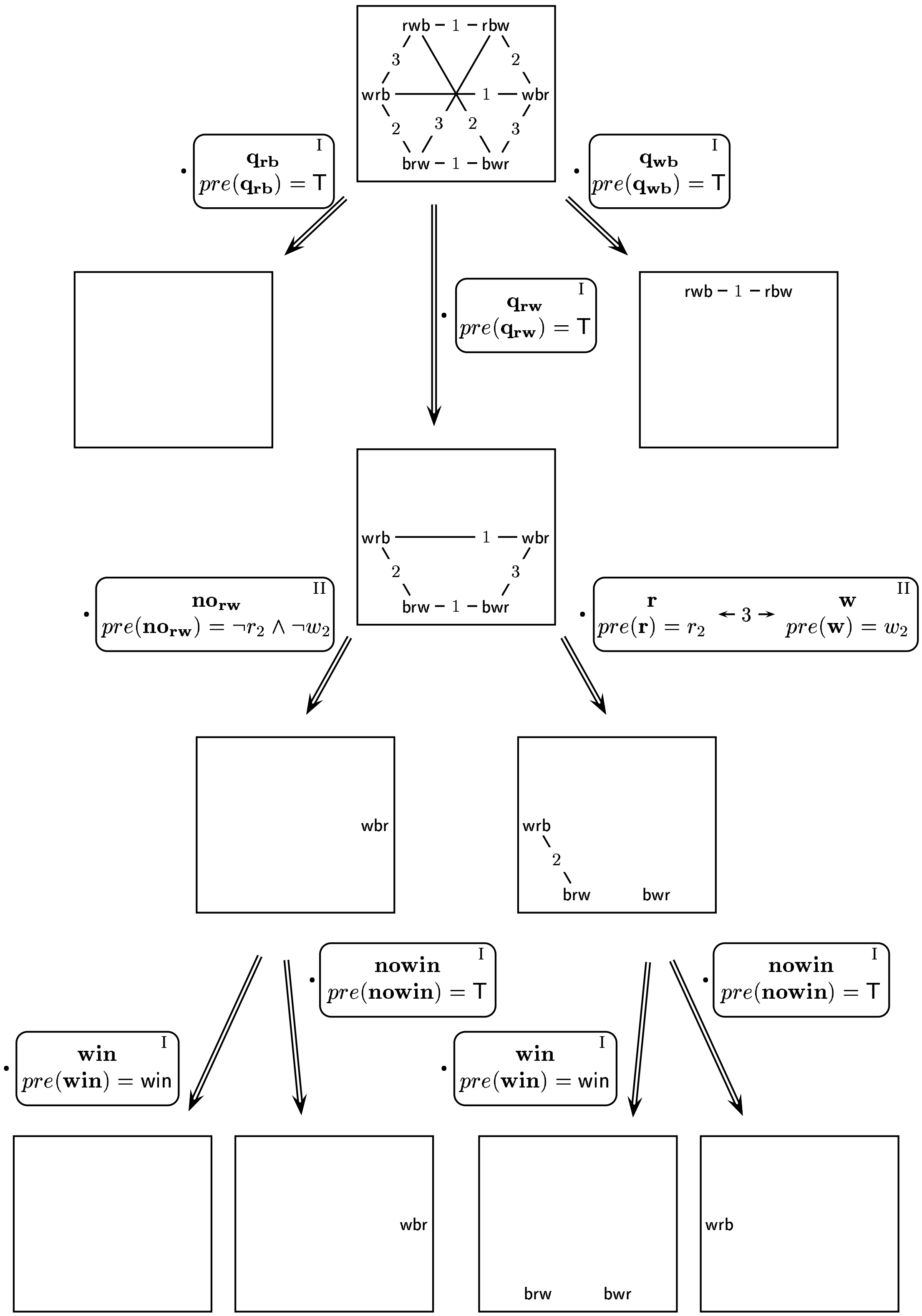

Figure 3.2: Strategic knowledge development in the game Hexas

Chapter 1 provides six examples of strategic knowledge development in the game Q-A. In chapter 2 we tried to model the strategic knowledge development by dynamic epistemic logic, but we found the analysis unsatisfying. In this section we analyze the the game and the knowledge development during the game, by modeling the game as a dynamic epistemic game and the application of •-execution to model the strategic knowledge development. First, we again give a short introduction to the game.

Description of Q-A The game consists of two players, a table and three cards. The cards are a red, a blue and a white card. Both players are dealt a card and the third card is put on the table, facing down. The players can only see their own card. After the dealing of the cards, player 1 asks player 2 a question. Player 1 has a choice of three possible questions: 1. "Are you holding the red card?"; 2. "Are you holding the white card?" and 3. "Are you holding the blue card?". Player 1 is the only player to ask a question. Then, player 2 answers player 1's question. Player 2 must answer the question of player 1 truthfully. When player 2 has answered player 1's question, both players make a guess about the card on the table and the card on the table is shown to the players. Players who guess correct win.

3.4.1 Game & Strategies

In this subsection, we model the game Q-A as a dynamic epistemic game. For each player, we determine what constitutes a strategy for the player.

Dynamic Epistemic Game Before we introduce the dynamic epistemic game, some notation. A deal of cards is denoted by sequence of three letters. rwb denotes player 1 is holding the red card, player 2 is holding the white card and the blue card is lying on the table. Let c_n and c_t respectively denote player n is holding card c and card c is lying on the table. Q-A is modeled as the dynamic epistemic game $\{N, M, H, P\}$:

- $N = \{1, 2\}$;
- $M = (W, R, V)$:
  - $W = \{\mathsf{rwb}, \mathsf{rbw}, \mathsf{wrb}, \mathsf{wbr}, \mathsf{brw}, \mathsf{bwr}\}$;
  - $R(1) = \{(\mathsf{rwb}, \mathsf{rbw}), (\mathsf{wrb}, \mathsf{wbr}), (\mathsf{brw}, \mathsf{bwr})\}$ and $R(2) = \{(\mathsf{rwb}, \mathsf{bwr}), (\mathsf{wrb}, \mathsf{brw}), (\mathsf{rbw}, \mathsf{wbr})\}$;
  - $V(\mathsf{rwb}) = \{r_1, w_2, b_t\}$, $V(\mathsf{rbw}) = \{r_1, b_2, w_t\}$, $V(\mathsf{wrb}) = \{w_1, r_2, b_t\}$, $V(\mathsf{wbr}) = \{w_1, b_2, r_t\}$, $V(\mathsf{wbr}) = \{w_1, b_2, r_t\}$, $V(\mathsf{brw}) = \{b_1, r_2, w_t\}$ and $V(\mathsf{bwr}) = \{b_1, w_2, r_t\}$.
- $H = \{\varnothing, (q_r), (q_w), (q_b), (q_r, no_r), (q_w, no_w), (q_b, no_b), (q_r, yes_r), (q_w, yes_w), (q_b, yes_b)\}$;
  - $q_r = (\{\mathbf{q_r}\}, \{(1, \emptyset)\}), (2, \emptyset)\})\}, \{(\mathbf{q_r}, \mathsf{T})\})$;
  - $yes_r = (\{\mathbf{yes_r}\}, \{(1, \emptyset)\}), (2, \emptyset)\})\}, \{(\mathbf{yes_r}, r_2)\})$;
  - $no_r = (\{\mathbf{no_r}\}, \{(1, \emptyset)\}), (2, \emptyset)\})\}, \{(\mathbf{no_r}, \neg r_2)\})$.

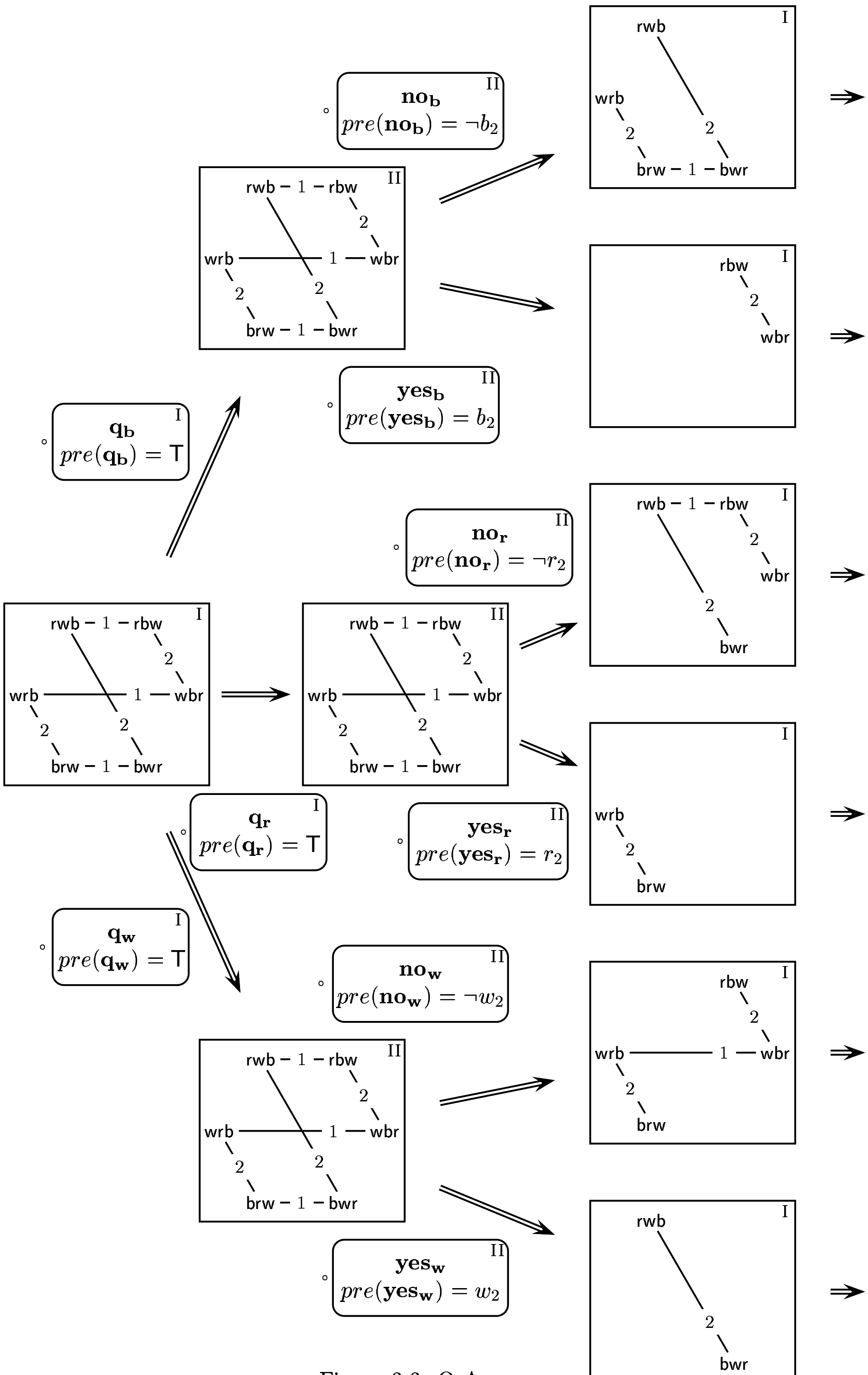
no_b
pre(no_b) = ¬b_2
yes_b
pre(yes_b) = b_2
q_b
pre(q_b) = T
no_r
pre(no_r) = ¬r_2
q_r
pre(q_r) = T
yes_r
pre(yes_r) = r_2
q_w
pre(q_w) = T
no_w
pre(no_w) = ¬w_2
yes_w
pre(yes_w) = w_2

Figure 3.3: Q-A

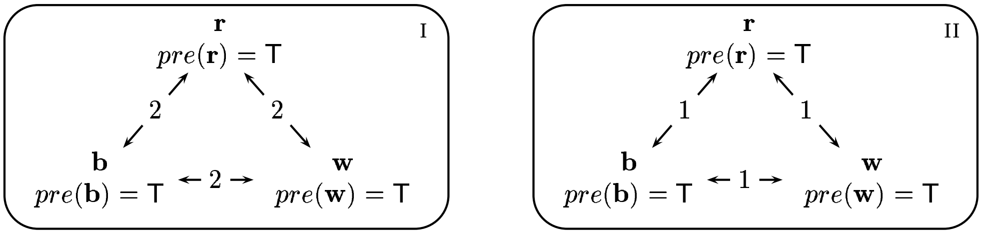

Figure 3.4: The action models $guess_1$ and $guess_2$

- $P(1) = \{\varnothing\}$ and $P(2) = \{(q_r), (q_w), (q_b)\}$.

The dynamic epistemic game is displayed in figure 3.3.

The dynamic epistemic game does not model the guesses of the players. Primarily, we are interested in the knowledge development during the question and answer. The histories of the dynamic epistemic game can be extended with action models modeling the guesses of the players. The guesses of player 1 are modeled by the action model $guess_1$:

$$guess_1 = (\{\mathbf{r}, \mathbf{w}, \mathbf{b}\}, \{(1, \emptyset), (2, \{(\mathbf{r}, \mathbf{w}), (\mathbf{r}, \mathbf{b}), (\mathbf{w}, \mathbf{b})\}, \{(\mathbf{r}, \top), (\mathbf{w}, \top), (\mathbf{b}, \top)\}).$$

The guesses of player 2 are modeled by the action model $guess_2$:

$$guess_2 = (\{\mathbf{r}, \mathbf{w}, \mathbf{b}\}, \{(1, \{(\mathbf{r}, \mathbf{w}), (\mathbf{r}, \mathbf{b}), (\mathbf{w}, \mathbf{b}), (2, \emptyset), \}, \{(\mathbf{r}, \top), (\mathbf{w}, \top), (\mathbf{b}, \top)\}).$$

The action models are displayed in figure 3.4.

Strategies A pure strategy of a player is an function prescribing the player which action to take in every game state in which the has to take an action. In game states which are indistinguishable to the player the player takes the same action. We have worked out a pure strategy of player 1 and a pure strategy of player 2. The strategies do not contain guessing actions, because the guesses are left out of the dynamic epistemic game.

A strategy of player 1 prescribes her which question to ask given the card player 1 is holding. As an example we have worked out a strategy s_1:

$$s_1 = \begin{cases} s_1(\varnothing)(\mathsf{rwb}) = s_1(\varnothing)(\mathsf{rbw}) = \mathbf{q_b} \\ s_1(\varnothing)(\mathsf{wrb}) = s_1(\varnothing)(\mathsf{wbr}) = \mathbf{q_w} \\ s_1(\varnothing)(\mathsf{brw}) = s_1(\varnothing)(\mathsf{bwr}) = \mathbf{q_r} \end{cases}$$

The strategy s_1 is such that player asks about the blue card when holding the red card, asks about the white card when holding the white card and asks about the red card if she holds the blue card.

A strategy of player 2 specifies which answers player 2 gives, given the question of player 1 and the card player 2 is holding. The answers player 2 can give are constrained by the preconditions of the answers. In every game state in which player 2 has to give an answer, only one answer is available. Player 2 has only one pure strategy s_2:

$$s_2 = \begin{cases} s_2(q_b)(\mathsf{rwb}, \mathbf{q_b}) = s_2(q_b)(\mathsf{bwr}, \mathbf{q_b}) = \mathbf{no_b} \\ s_2(q_b)(\mathsf{rbw}, \mathbf{q_b}) = s_2(q_b)(\mathsf{wbr}, \mathbf{q_b}) = \mathbf{yes_b} \\ s_2(q_b)(\mathsf{wrb}, \mathbf{q_b}) = s_2(q_b)(\mathsf{brw}, \mathbf{q_b}) = \mathbf{no_b} \\ \\ s_2(q_r)(\mathsf{rwb}, \mathbf{q_r}) = s_2(q_r)(\mathsf{bwr}, \mathbf{q_r}) = \mathbf{no_r} \\ s_2(q_r)(\mathsf{rbw}, \mathbf{q_r}) = s_2(q_r)(\mathsf{wbr}, \mathbf{q_r}) = \mathbf{no_r} \\ s_2(q_r)(\mathsf{wrb}, \mathbf{q_r}) = s_2(q_r)(\mathsf{brw}, \mathbf{q_r}) = \mathbf{yes_r} \\ \\ s_2(q_w)(\mathsf{rwb}, \mathbf{q_w}) = s_2(q_w)(\mathsf{bwr}, \mathbf{q_w}) = \mathbf{yes_w} \\ s_2(q_w)(\mathsf{rbw}, \mathbf{q_w}) = s_2(q_w)(\mathsf{wbr}, \mathbf{q_w}) = \mathbf{no_w} \\ s_2(q_w)(\mathsf{wrb}, \mathbf{q_w}) = s_2(q_w)(\mathsf{brw}, \mathbf{q_w}) = \mathbf{no_w} \end{cases}$$

Definite Knowledge During the game, the players develop definite knowledge about the deal of cards, as the player 2 answers the question of player 1. The definite knowledge development of the players is directly modeled by the game models and action models of the dynamic epistemic game. In chapter 2, we analyzed the definite knowledge development of the players if player 1 asks about the red card in dynamic epistemic logic. The analysis resulted in a sequence of Kripke models, modeling the definite knowledge development of the players in the game Q-A. The Kripke models are displayed in figure 2.5. The Kripke models are equal to the resulting game models after the action model q_r. The game models correctly model the definite knowledge development of the players.

3.4.2 Strategic Knowledge

In chapter 1 we have given several examples of how the players of the game can develop strategic knowledge, given the question of player 1. In this subsection we will analyze the strategic knowledge the players develop by the dynamic epistemic game and •-execution.

In every example provided by chapter 1, player 1 asks player 2 about the red card and player 2 does not hold the red card. For every example we first formalize the strategy of player 1 and then apply •-execution to calculate the strategic knowledge of the players, if possible.

Always ask your own card Player 1 always asks about the card she is holding and this is common knowledge between the players. Player 1 uses the pure strategy s_1:

$$s_1 = \begin{cases} s_1(\varnothing)(\mathsf{rwb}) = s_1(\varnothing)(\mathsf{rbw}) = \mathbf{q_r} \\ s_1(\varnothing)(\mathsf{wrb}) = s_1(\varnothing)(\mathsf{wbr}) = \mathbf{q_w} \\ s_1(\varnothing)(\mathsf{brw}) = s_1(\varnothing)(\mathsf{bwr}) = \mathbf{q_b} \end{cases}$$

The strategic knowledge development of the players if it is common knowledge player 1 and player 2 use the pure strategies s_1 and s_2 respectively is displayed in figure 3.5.

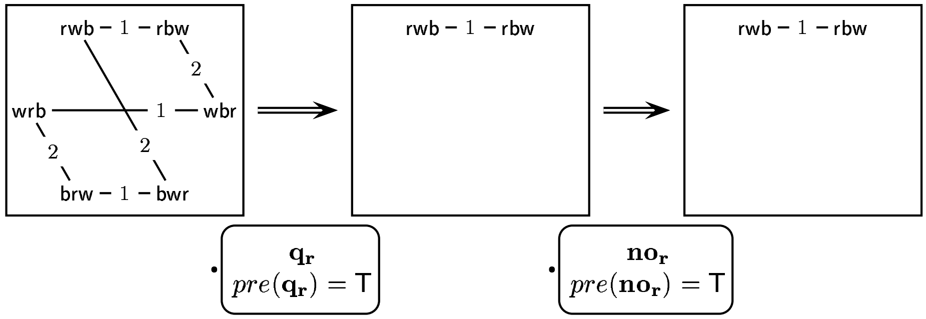

Figure 3.5: Always ask your own card

Always ask the red card It is common knowledge player 1 always asks player 2 about the red card, independent of the card player 1 is holding. The strategy of player 1 is s_1:

$$s_1 = \begin{cases} s_1(\varnothing)(\mathsf{rwb}) = s_1(\varnothing)(\mathsf{rbw}) = \mathbf{q_r} \\ s_1(\varnothing)(\mathsf{wrb}) = s_1(\varnothing)(\mathsf{wbr}) = \mathbf{q_r} \\ s_1(\varnothing)(\mathsf{brw}) = s_1(\varnothing)(\mathsf{bwr}) = \mathbf{q_r} \end{cases}$$

The strategic knowledge the players develop as they commonly know player 1 always asks about the red card, i.e. uses strategy s_1, and player 2 uses his pure strategy s_2 is displayed in figure 3.6. The players do not learn anything about the question, the Kripke model is unchanged. Furthermore the players do not learn extra information by the answer of player 2. Players do not develop extra knowledge (strategic knowledge) if the players commonly know player 1 always asks about the red card.

Never ask your own card Player 1 uses the following strategy:

$$s_1 = \begin{cases} s_1(\varnothing)(\mathsf{rwb}) = s_1(\varnothing)(\mathsf{rbw}) = \mathbf{q_w} \\ s_1(\varnothing)(\mathsf{wrb}) = s_1(\varnothing)(\mathsf{wbr}) = \mathbf{q_r} \\ s_1(\varnothing)(\mathsf{brw}) = s_1(\varnothing)(\mathsf{bwr}) = \mathbf{q_w} \end{cases}$$

Player 1 asks about the white card when holding the red or blue card and asks about the red card when holding the white card. The strategy is such that

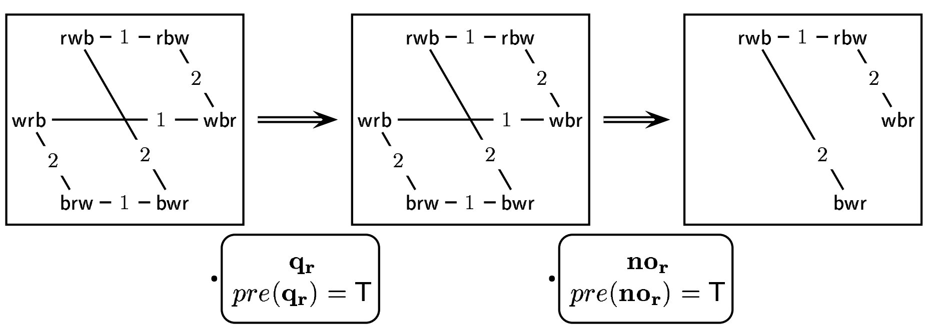

Figure 3.6: Always ask the red card

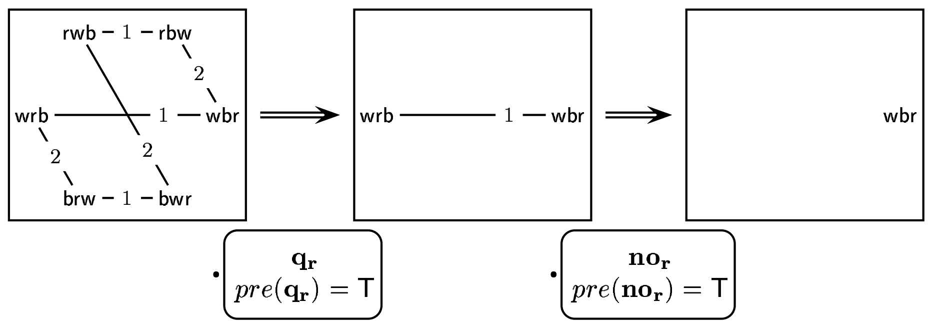

Figure 3.7: Never ask your own card

player 1 never asks about her own card. The strategic knowledge development of the players when it is common knowledge player 1 uses the above strategy is displayed in figure 3.7.

Player 1 informs player 2 about which card she is holding, when asking about the red card, as player 2 knows player 1 only asks about the red card when holding the white card. Player 2 knows the deal cards before player 1. When player 2 has responded he is holding the red card, it becomes common knowledge wbr is the deal of cards.

Strict Domination If the players are rational, player 1 will not ask about the red card when holding the red card. The pure strategies of player 1, such that she asks about her own card, when holding the red, white or blue card are strictly dominated. If it is common knowledge the players are rational, player 2 learns player 1 is not holding the red card, when she asks about the red card. We call this knowledge rational knowledge, because it is developed when players commonly know all players are rational. Rational knowledge is an instance of strategic knowledge.

Player 1 and 2 develop this knowledge, because they know player 1 will not use strictly dominated pure strategies. We cannot model the strategic knowledge players develop, if the players commonly know which pure strategies will *not* be used. Using •-execution, we can only model strategic knowledge players develop when the players commonly know which pure strategies will be used by the players.

Figure 3.8 displays how the rational players develop knowledge, when the players commonly know all players are rational. These Kripke models were not calculated by •-execution, but were determined by hand, this case cannot be modeled using •-execution.

Nash-equilibrium When the players play the game Q-A according to a Nash-equilibrium, the players randomize over their pure strategies. The players use mixed strategies. A Nash-equilibrium of the game Q-A is such that player 1 randomly asks about a card she is not holding. If the players commonly know the game is played according to this Nash-equilibrium, the players learn which card player 1 is not holding when player 1 asks her question.

We cannot model the rational knowledge the players develop using •-execution as we cannot handle the mixed strategies the players use when playing according

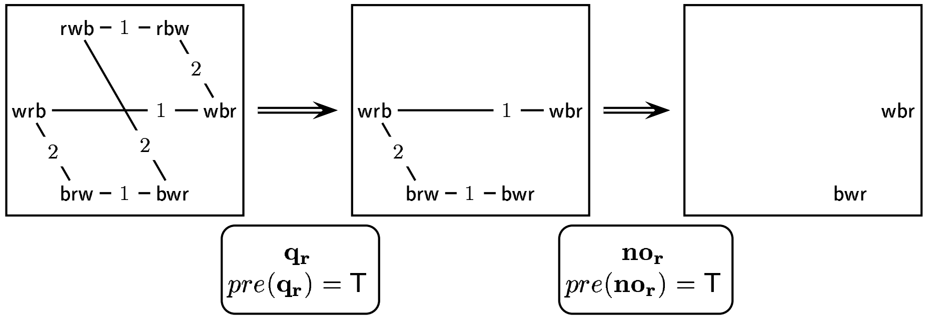

Figure 3.8: Rational knowledge in the game Q-A.

to a Nash-equilibrium. The rational knowledge the players develop is equal to the knowledge the players develop, when it is common knowledge player 1 does not use a strictly dominated strategy.

Uncommon Knowledge Let it be the case player 2 has learned through some event that player 1 uses the pure strategy such that player asks the white card when holding the blue and asks the red card when holding the red or white card. Player 1 does not know player 2 has learned that she is using this strategy. When player 1 asks player 2 about the red card, player 2 learns player 1 is not holding the blue card. Player 1 still assumes player 2 is ignorant about the deal of cards, while in fact if player 2 is not holding the blue card, he knows the deal of cards.

We cannot model the knowledge player 2 develops using •-execution. We can only model the cases in which the players commonly know which pure strategies are used by the players.

3.4.3 Conclusions of the Card Game

We have analyzed the game Q-A by modeling the game as an dynamic epistemic game. The analysis primarily focuses on the knowledge development of the players. The definite knowledge the players develop in the game is modeled by the dynamic epistemic model itself.

Chapter 1 provided 6 examples of strategic knowledge development in the game Q-A. Our goal was to model the strategic knowledge the players develop. By modeling the game Q-A as a dynamic epistemic game and the application of •-execution we can model strategic knowledge if and only if the players use pure strategies and have common knowledge about the used pure strategies. If a player uses a mixed strategy, or if it is not commonly known which strategies are used, we cannot model the strategic knowledge the players develop by the application of •-execution.

3.5 Discussion

Game actions & Knowledge actions We used Kripke models and action models as building blocks to model games. The result is a dynamic epistemic game. The Kripke models and action models are used as building blocks, because we want to model the knowledge development of players and the strategies of players in one game model.

The actions of an action model are actions which alter the knowledge of the players. This is what we call knowledge actions. In dynamic epistemic games, actions of an action model also model game actions. Game actions are actions taken in a game by players. In dynamic epistemic games, knowledge actions become game actions and vice versa.

In dynamic epistemic games, action of an action model are differently viewed, than in dynamic epistemic logic. In dynamic epistemic logic, actions are events which just happen to occur. In dynamic epistemic game, the action of an action model now models an action initiated by a player. The modeling knowledge development in games is different. In *Knowledge Games* [20], Van Ditmarsch models a question and answer as one action. In dynamic epistemic games, we cannot model the question and answer as one action, as the question and answer are actions initiated by different players. The question and answer are modeled as two actions.

We find the modeling of the question and answer as two separate actions more elegant and natural. The knowledge distribution of the players directly after the question, but before the answer, is relevant with respect to strategic knowledge and cannot be kept unmodeled. By separately modeling the question and answer, the knowledge distribution directly after the question is modeled.

The state of a game has become a state of a Kripke model. The accessibility relations serve two purposes. First of all, the accessibility relations determine the knowledge of the players over the states. Secondly, the accessibility relations constrain the strategies of the players, just as information sets in a game of imperfect information.

Modeling The modeler of a dynamic epistemic game has to determine the action models and Kripke model. We find the modeling of a game of imperfect information as a dynamic epistemic game easier than as the game theoretic model, because a dynamic epistemic game is modular. The modeler only has to model the separate action models and a Kripke model.

Games of imperfect information can be modeled as a dynamic process by a dynamic epistemic game. By attributing information about the knowledge development induced by an action to an action model, the knowledge development and indistinguishability relations can be calculated automatically. This makes the dynamic modeling of a game of imperfect information is more elegant.

A tree representing a dynamic epistemic game does not contain connections between game models, in contrast with the game trees of game theoretic models. Every node in the tree can be viewed as a subgame, which makes the trees and games easier and better to comprehend.

Constraints Dynamic epistemic games have a price: not every game, which can be modeled by game theory, can be modeled as a dynamic epistemic game. We have given five constraints which must hold for a game to be modeled as a dynamic epistemic game. A game cannot be modeled as a dynamic epistemic game, if one of these five constraints does not hold.

Knowledge Dynamic epistemic games model the definite knowledge players develop during a game. The definite knowledge development is modeled dynamically, just as dynamic epistemic logic. For every state of the game, the knowledge of the players is also determined. This gives more insight to a game, as it directly models the definite knowledge players develop.

Given a dynamic epistemic game as a basis, we can model strategic knowledge. Strategic knowledge is calculated by •-execution. We add the used strategy profile as extra information to calculate the strategic knowledge of the players. The action models itself only model the definite knowledge development of the players. Strategic knowledge gives even extra insight to a game of imperfect information, next to definite knowledge, as the knowledge of the players, which consists for a large margin out of strategic knowledge, is completely modeled. Up to now we can only model one specific form of strategic knowledge. We are tied to situations in which players commonly know the used strategies by the players and the players use pure strategies.

3.6 Further Research

Payoffs & Rationality Dynamic epistemic games lack payoffs. A payoff function would assign payoffs to states of terminal game models. Furthermore mixed strategies need to be introduced. If payoffs and mixed strategies are given, prolonged game theoretical analysis is possible. For a dynamic epistemic game, the definitions of strict domination, Nash-equilibrium, subgame-perfect equilibrium, assessment equilibrium are subject of further research.

Strategic Knowledge Strategic knowledge can be modeled if the players commonly know which strategies are used. This idea of strategic knowledge is inspired by *Opponent-Model Search* by Iida et al. [12, 14, 13, 11] and research on opponent modeling by Donkers [6, 7, 8] and Carmel and Markovitch [3, 4, 5]. In opponent-model search, a player assumes his opponent uses a certain algorithm to determine the moves he makes. The information about the used algorithm is summarized in an opponent model. The opponent model is used to determine the moves for the player himself. In opponent model search, assumptions about used strategy by the opponent is used to determine a strategy. In dynamic epistemic games, knowledge about the used strategies is used to develop knowledge about the state of the game.

The strategic knowledge we model, given a dynamic epistemic game and •-execution, is just a fraction of what we understand as strategic knowledge. In further research, the possibilities to model other forms of strategic knowledge, given as examples, need to be investigated.

Given a dynamic epistemic game and •-execution, we can model the strategic knowledge players develop if it is common knowledge which strategies the players use. We would also like to model other forms of strategic knowledge. Our idea is to model the strategic information players have about the used strategies as Kripke models. The Kripke model would model the knowledge the players have about the used strategy. The strategic knowledge of the players could be calculated by some execution of this Kripke model with a game model. This can be a general method to calculate strategic knowledge, and applicable for several forms of strategic knowledge.

Game Tree Search The trees representing dynamic epistemic games do not have connections between nodes, such as information sets. Every node can be viewed as a subgame. It seems to us that some form of backward induction is possible, using dynamic epistemic games. We will investigate the possibilities to search a dynamic epistemic game tree by methods similar to backward induction, Zermelo's algorithm and alpha-beta search.

3.7 Conclusions

We have introduced dynamic epistemic games. Games of imperfect information can be modeled easier and more elegant as a dynamic epistemic game than the game models provided by game theory. Unfortunately, we cannot model every game which can be modeled by game theory as a dynamic epistemic game. The constraints on a dynamic epistemic game, limit the games we can model.

A dynamic epistemic game also models the definite knowledge players develop in a game. Strategic knowledge can be model on the basis of a dynamic epistemic game. Up to now, we can only model the strategic knowledge players develop if it is common knowledge which strategies the players use. Dynamic epistemic games can also be used as a basis on which we model other forms of strategic knowledge.

Dynamic epistemic games are introduced to model strategic knowledge. The model gives insight to the knowledge players develop during a game. On this knowledge the strategies of the players are determined. Furthermore, the model seems to have other applications as game tree search and game theoretic analysis.

Conclusion

In games of imperfect information players are uncertain about the precise state of the game. Players learn about the state of the game by explicit information exchange and by observing actions taken by other players. We call the knowledge players develop about the state of the game through explicit information exchange definite knowledge. The taken actions depends on the strategies of the players and the state of the game. A player develops knowledge about the game state by observing actions taken by other players, if the player has knowledge about the used strategies of the opponents. We call this form of knowledge strategic knowledge.

Game theory provides a game model to model games of imperfect information. Given a model of the game of imperfect information, the available strategies to the players can be determined. A game can be analyzed to determine what constitutes rational behavior. A rational player chooses her actions such that her expected payoff is maximized. If a rational player has to take actions under uncertainty, the player has in mind an expectation about the uncertainty. When observing the actions taken by her opponents, a rational player develops rational knowledge about the state of the game.

Q-A is an example of a game of imperfect information. Q-A consists of three cards, two players and a table. The cards are dealt over the players and the table. After the dealing of the card, player 1 can ask player 2 three questions. She can ask whether he is holding the red card, whether he is holding the white card and whether he is holding the blue card. The goal of the players is to correctly guess the card on the table at the end of the game. In the game Q-A it is irrational for player 1 to ask player 2 about the card player 1 is holding herself. If player 2 knows player 1 is rational, he learns after player 1's question that player 1 is not holding the card she asks player 2 about.

In dynamic epistemic logic as developed by Hans van Ditmarsch, knowledge and knowledge development can be modeled. Definite knowledge and its development in games of imperfect is elegantly modeled by dynamic epistemic logic. During the game, Kripke models model the knowledge distribution about the state of the game over the players. The explicit information exchanges are modeled by action models. The knowledge development is calculated by the execution of the action models.

In dynamic epistemic logic, strategic knowledge and its development is not modeled to satisfaction. If a player has knowledge about the used strategies, a player develops strategic knowledge by observing the actions taken by her opponents. To model this form of knowledge development, the taken actions need to be translated into an action model. The interpretation of the taken actions, depends on the knowledge of the players about the used strategies. In general, it is too hard to determine how the knowledge about the used strategies determines the interpretation of the actions in a game of imperfect information.

The main issue of this master thesis is the introduction of dynamic epistemic games. In dynamic epistemic games, games of imperfect information are modeled by Kripke models and action models. In dynamic epistemic games, the definite knowledge players develop during the game is directly modeled by the Kripke models and action models. The dynamic epistemic game also determines the available strategies to the players. In dynamic epistemic games, knowledge and strategies are connected formally, which is necessary to model strategic knowledge.

If the players use pure strategies and it is common knowledge which strategies are used, strategic knowledge is modeled by dynamic epistemic games and the application of $\bullet$-execution. Cases of strategic knowledge, such that the players do not commonly know what the used strategies are, or the players do not use pure strategies cannot be modeled by the application of $\bullet$-execution. To model others forms of strategic knowledge, we speculate that the knowledge of the players about the used strategies needs to be modeled in a general form. Given this general form, strategic knowledge could be calculated.

Bibliography

[1] A. Baltag. A logic for suspicious players: Epistemic actions and belief-updates in games. *To appear in Bulletin of Economic Research*, 2000.

[2] K. Binmore. *Fun & Games*. D. C. Heath, Lexington MA, 1992.

[3] D. Carmel and S. Markovitch. The M* algorithm: Incorporating opponent models into adversary search. Technical report, Technion, March 1994. Technical Report CIS report 9402.

[4] D. Carmel and S. Markovitch. Incorporating opponent models into adversary search. In *Proceedings of the Thirteenth National Conference on Artificial Intelligence*, Portland, OR, 1996.

[5] D. Carmel and S. Markovitch. Opponent modeling in multi–agent systems. In G. Weiß and S. Sen, editors, *Adaptation and Learning in Multi–Agent Systems*, pages 40–52. Springer-Verlag: Heidelberg, Germany, 1996.

[6] H. H. L. M. Donkers, J. W. H. M. Uiterwijk, H. Iida, and H. J. van den Herik. β-pruning and opponent-model search. Technical Report CS 00-05, Technical Reports in Computer Science., 2000.

[7] H. H. L. M. Donkers, J. W. H. M. Uiterwijk, and H. J. van den Herik. Admissibility in opponent-model search. Accepted for 13th Belgian-Dutch Conference on Artificial Intelligence (BNAIC'01), 2001.

[8] H. H. L. M. Donkers, J. W. H. M. Uiterwijk, and H. J. van den Herik. Probabilistic opponent-model search. *Information Sciences*, 135:123–149, 2001.

[9] R. Fagin, J. Halpern, Y. Moses, and M.Vardi. *Reasoning about Knowledge*. MIT Press, 1995.

[10] J. Gerbrandy and W. Groeneveld. Reasoning about information change. *Journal of Logic, Language an Information*, 6:147–196, 1997.

[11] H. Iida, Y. Kotani, J. W. H. M. Uiterwijk, and H. J. van den Herik. *Advances in Computer Chess*, volume 8, chapter Gains and Risks in OM Search, pages 153–166. Universiteit Maastricht, Department of Computer Sciences, Maastricht, The Netherlands, 1997.

[12] H. Iida, J. W. H. M. Uiterwijk, and H. J. van den Herik. Opponent-model search. Technical Report in Computer Science 93-03, University of Limburg, Maastricht, 1993.

[13] H. Iida, J. W. H. M. Uiterwijk, and H. J. van den Herik. A game-tree model including an opponent model. In J.-J. Meyer and L. van der Gaag, editors, *Proceedings of the Eight Dutch Conference on Artificial Intelligence (NAIC '96)*, pages 223–234, 1996.

[14] H. Iida, J. W. H. M. Uiterwijk, H. J. van den Herik, and I. S. Herschberg. *Thoughts on the Application of Opponent-Model Search*, pages 61–78. University of Limburg, Dept. of Computer Science, Maastricht, The Netherlands, 1994.

[15] B. Kooi, H. van Ditmarsch, and W. van der Hoek. Action language. No note, July 2001.

[16] J.-J. Meyer and W. van der Hoek. *Epistemic Logic for AI and Computer Science*. Cambridge University Press, 1995.

[17] M. J. Osborne and A. Rubinstein. *A Course in Game Theory*. MIT Press, 1994.

[18] A. Rubinstein, 2001. Personal Communication.

[19] M. Sakuta and H. Iida. Solving kriegspiel-like problems: exploiting a transposition table. *International Computer Games Association Journal*, 23(4):218–230, December 2000.

[20] H. P. van Ditmarsch. *Knowledge Games*. PhD thesis, University of Groningen, The Netherlands, 2000.