Department of Computer Science, University of Otago

Technical Report OUCS-2004-02

Logic and game theory of Pit

Author:
Hans van Ditmarsch
Department of Computer Science

Status: completed manuscript

Department of Computer Science,
University of Otago, PO Box 56, Dunedin, Otago, New Zealand

http://www.cs.otago.ac.nz/trseries/

Logic and game theory of Pit

Hans van Ditmarsch[*][†] 19 March 2004

Computer Science, University of Otago, New Zealand, hans@cs.otago.ac.nz

Abstract

Pit is a multi-player card game that simulates the commodities trading market, and where actions consist of bidding and of swapping cards. We define a simplification of that game for which we present a detailed description of all dynamical game features. The description is in a standard language for dynamic epistemics. This formalization is then used to outline the game theory for a simplification of the Pit game. This uncovers some interesting equilibria.

1 Introduction

Pit is a multi-player card game where actions consist of swapping cards. The first player to declare a certain hand of cards wins the game. From a different theoretical point of view, various ramifications involving the Pit game have been investigated [5, 7]. The former uses Pit to illustrate the supply and demand cycle in the general economics classroom. The latter two may be seen as a study in requirements engineering for electronic market simulations. In this paper we address the logical dynamics of the game and also present some game theoretical results. The starting point to specify the logical dynamics is the language presented in [8], which forms part of an ongoing line of research in logical dynamics [4, 1], and which has specifically been used to describe (other) card game actions in [9]. Another starting point for the logical dynamics are the card game state descriptions originally presented in [11].

The structure of this paper is as follows. In Section 2 we describe the Pit game, and in Section 3 some abstractions from the real game that facilitate formalization. Given that, we then give an overview of all conceivable game actions and game states in Section 4. This includes a new action feature 'assignment', essential to describe trading cards. We illustrate the dynamics for the case of three players and three resources of each two cards: the SixPit game. In Section 5 we make a small exploration into the game theory of Pit, only for the

[*]I kindly acknowledge useful comments from Miklos Szilagyi.

[†]This work is related and indebted to a collaboration involving Pit between the author, Johan Lövdahl (Computer Science, Linköbing and Otago University), and Stephen Cranefield and Martin Purvis (Information Science, Otago University).

specific case of SixPit. That uncovers three equilibria, plus a nonstable profile that benefits two collaborating players. The last appears to be an interesting 'chicken'-like dilemma in an e-commerce setting.

There is a slight discrepancy between 'game actions' as modelled in logic and 'game actions' as abstract parameters in game theory, and similarly between 'game states' in logic or in game theory. Instead of introducing different terminology, we will rely on the reader's perception of the context of such terms and we will only point out marked differences when relevant.

2 Pit game

The essentials of the Pit game are taken from

`www.hasbro.com/common/instruct/pit.pdf`

> *The object of Pit is to corner the market on Barley, Corn, Flax, Hay, Oats, Rye and Wheat by trading cards with other players. Pit can be played by three to seven players. There are nine cards in each suit. If three play, use only three complete suits. If four play, use four complete suits, etc. Use the complete Pit deck for seven players. Place the trading bell in the center of the table and select one player to shuffle the deck and deal out nine cards to each player. The Dealer should allow the players 30 seconds to sort their cards and decide mentally on which commodity (Wheat, Rye, Oats. etc.) they will attempt to corner. Players should try to corner the commodity of which they hold the most cards. When the cards have been sorted. the Dealer strikes the bell and announces, "The Exchange is open." Then, any player may begin to trade cards with other players. To trade, he takes from his hand one to four cards of the same suit, holds the cards up so that the suits do not show and calls out, "Trade One! One! One!" or "Two! Two! Two!" or "Three! Three! Three!" or "Four! Four! Four!" depending on the number of cards being traded. Players continue calling out their numbers until the cards have been exchanged. If a player wishes to exchange cards with another player, he must call in return, "One! One! One!", "Two! Two! Two!", etc. and trade an equal number of cards of the same suit with that player. If a player wishes to trade three or four cards and other players will only exchange lower numbers, he may drop his bid and trade the smaller number of cards. Trading continues until one player gets nine cards of the same suit. That player must quickly ring the bell and call out, "Corner on Wheat!" (or whatever the commodity may be). The player then scores the amount marked on the commodity he has cornered (Wheat, 100 points; Oats, 60 points, etc.) and records this on the score pad. When a corner is won, all the cards are reshuffled and dealt by the last winner and another corner is played for. The game is won by the first trader to get 500 points.*

3 Abstraction of Pit

An important aspect of the Pit game for multi-agent systems researchers may well be its real-time asynchronous and concurrent features: players act under pressure, using incomplete or incompletely processed information, and with also otherwise restricted rationality. It is this mix of realistic agent behaviour in an otherwise highly procedural game setting that appears to make it a suitable vehicle to simulate trading and negotiation [7]. Our current more restricted interest is in the logical dynamics and game theory of Pit. For that, we even have to make further abstractions from the reality of live agents. Specifically, we need to abstract from asynchronous features of Pit, and we assume 'ideal agents', i.e. perfectly rational and perfectly logical agents. Of course, it may then be occasionally unclear how the results we will obtain for this abstraction are relevant for 'the real game': a real pitfall for a multi-agent systems researcher.

We consider the following abstraction of the Pit game:

Given is a set of players (agents) N and a natural number m. We consider *Pit* games for players N and a deck of cards consisting of $|N|$ suits, i.e. equal to the number of players, and with m cards per suit. The set of suits is called U and the set of all cards Q. Nondeterministically determine a deal of the cards over the players, where each player gets m cards. Given a player's hand, the set of possible offers for trade consists of any number of cards of the same suit in that player's hand. All players each make one offer, simultaneously. Choose nondeterministically one from the matching offers (i.e., offers for the same number of cards), and execute it. Repeat this 'make-offer'/'execute-trade' process until at least one player can corner the market. Choose one from the achieved corners nondeterministically. This player wins the game.

This abstraction is a significant departure from the real game. The act of accepting an offer pending the still uncertain execution of that trade is not considered an action in the abstraction, in other words: we ignore a certain asymmetry between the agents in the process leading up to execution of a trade. Also, in this abstraction, after a trade, all non-honoured offers are considered withdrawn and do not remain valid. In the real Pit game, such offers remain valid. This means that a consecutive trade can take place that does not depend on the current game state but still on the previous game state: a complication with game theoretical consequences that *could* have been modelled but that we have chosen not to. In this abstraction, who is allowed to trade depends on chance, and not on choice. In the real game a player can *refuse* to trade with another player who makes a matching offer. This determines the cards a player is willing to offer to trade: yet another game theoretically relevant consequence, that we have not incorporated in the current abstraction.

Further generalizations are also conceivable. The number of agents may be different from the number of suits, and/or the number of cards in a suit may be less than the number of cards in a player's hand. We do not model those here.

We assume that all cards have equal value. This assumption is not required to describe Pit's dynamics, but is helpful to simplify its game theory. For the same reason, repetitions of the game, that are required in order to achieve the '500 points', are not considered.

SixPit A running example in the continuation will be the Pit game for three players Anne, Bill, and Cath (a, b, and c) that each hold two cards from a pack consisting of two Wheat, two Flax, and two Rye cards. A commodity corresponds to a suit. The suits are abbreviated as w, x, y. For the card deal where Anne holds a Wheat and a Flax card, Bill a Wheat and a Rye card, and Cath a Flax and a Rye card, we write $wx.wy.xy$, etc. The representation of game actions will be considered in the next section. For the purpose of modelling only, which requires rigid designators (object identities), we further assume that the cards are named $w^1, w^2, x^1, x^2, y^1, y^2$. We call this the SixPit game.

4 Dynamic epistemics of Pit

Given the abstractions, we can describe all game states and game actions formally. The initial state of the game, which is 'merely' a deal of cards where all players only know their own hand of cards, has been exhaustively explored in [11]. Most game actions can be modelled in a known language and semantics for logics of knowledge change [9, 10]. Their description is already worthwhile to report on, as a case-study in knowledge specification. But some action features need an extension of this language that has not been presented before. This we consider therefore of additional, independent, interest. The description of the initial game state and the description of the game actions fully determines what is known by arbitrary players in arbitrary game states.

We start with an overview of the knowledge of the players in this game and how this changes as a result of game actions. Before dealing the cards, it is common knowledge how many suits there are and how many cards there are in each suit. The following game actions (and *only* those actions) have epistemic content:

- *Dealing the cards to the players*
  As a result of this, it is common knowledge to all players: how many cards each player has, and that a player only knows his own cards. This allows for descriptions such as "player n knows that player n' does not know that n holds card q".

- *Offering i cards for trade*
  The offering player thus publicly announces that he has at least i cards of at least one commodity. The cards being offered are not made public.

- *Trading i cards*
  This is a 'semi-public announcement': a non-deterministic action that all agents know to take place (partly needed for synchronization purposes),

even though the observable part of the action may vary for different subgroups of all agents. In the case of trading cards, the two trading players gain common knowledge of the (current, switched) ownership of the swapped (sets of size i of) cards, whereas all players gain common knowledge that those two players have traded i cards. Also, a non-trading merely observing player, who knew one of the trading players to hold some given card, may now no longer know that.

- *Not cornering the market*
  When neither of the players that just traded announces a corner, this is indirectly 'observed' by the remaining players because 'the game goes on'. For our modelling purposes this is a public announcement that the trading players 'cannot win', i.e., that neither of them holds a complete suit.

- *Cornering the market*
  One of the players that just traded is the first to announce a corner in some commodity (i.e., that he holds a complete suit).

We emphasize once more, that there are no other dynamic features of the game, given the abstraction. We continue with the formalization of this epistemic dynamic setting. After the cards have been dealt, the game state results [11] that represents an arbitrary distribution of cards over players. This is a finite relational structure consisting of a domain of abstract points and a number of unary and binary relations between those points, plus a designated point. This may also be known as a pointed Kripke model. We give no details.

On such a structure we can interpret epistemic statements as in the examples above. For this, a 'simple' propositional language is sufficient. This has the advantage that such statements are then decidable, so that they can, in principle, be computed using various proof and/or model checking tools.

We choose some set of atoms and define various useful propositions. Atomic propositions q_n describe that player n holds card q. For example, the fact that Anne holds card w^1 is described by the atomic proposition w^1_a (one can read this as $(w^1)_a$). We introduce propositional connectives in standard ways[3]. Propositions u_n^{+i} stand for "player n holds at least i cards of commodity u" and are defined in the obvious way, e.g.:

$$\begin{array}{lcl} u_n^{+1} & :\Leftrightarrow & \bigvee_{i=1\dots m} u_n^i \\ u_n^{+m} & :\Leftrightarrow & \bigwedge_{i=1\dots m} u_n^i \end{array}$$

Instead of u_n^{+1} we also write u_n ('player n holds – at least – a u card') and instead of u_n^{+i} we also write $u...u_n$ (length i sequence of u) ('player n holds – at least – a i u cards'). Note that u_n^{+m} stands for 'player n holds exactly m cards of commodity u', as m is the number of cards in a player's hand. We also informally extend the notation $u...u_n$ to cover holding at least i cards of different suits. For example, if each player holds nine cards, as in the full version of the Pit game, $xxxxyy_a$ means that Anne holds at least four Flax and two

Rye cards. We let $Corner(n, u)$ (player n can corner the market in commodity u) stand for u_n^{+m}, and define $Corner(n) :\Leftrightarrow \bigvee_{u \in U} Corner(n, u)$ (player n can corner the market).

For example, w_a describes that Anne holds a Wheat card, and $Corner(a, w)$, or ww_a, that Anne corners the market on Wheat, i.e., "that she has at least two Wheat cards". And wx_a describes that that Anne's hand of cards consists of one Wheat and one Flax card.

Epistemic modal operators such as K_n, for individual knowledge, and C_N, for (public) common knowledge, are also introduced in standard ways[3]. For example, $K_a w_a$ stands for "Anne knows (i.e., sees) that she holds a Wheat card", and $\bigvee_{u=w,x,y} C_{abc}(u_a \rightarrow K_a u_a)$ stands for "It is common knowledge to all players that Anne knows the suit of the card that she is holding." All background knowledge takes this form of commonly known propositions. The two example propositions are indeed true for this application, but we have to be careful in these matters. Note the difference between *owning* a card – which we stipulate to be already true if it lies facedown in front of you, and *knowing* that you own/hold a card – only true after you've picked it up. In our case their is also a striking difference between knowing the suit of a card and knowing the identity of a card. Typically, $w_a^1 \rightarrow K_a w_a^1$ is false: Anne can determine if she is holding a Wheat card, by looking at it, but not *which* of the two Wheat cards that is. She cannot see if she is holding 'the card named w^1'. That is an abstraction only made in order to allow object identities. It is not supposed to show on the cards themselves.

The game actions can now be described as *epistemic actions* in an action language as in [9, 1]. In such languages, epistemic actions induce *information state transformations*, i.e. they induce binary relations between the pointed relational structures introduced before. For motivation and the formal semantics, see [9, 10]. Actions are interpreted as state transformers, and this corresponds in the language to a dynamic modal operator. We explain this by example only. Suppose the deal of cards is $wx.wy.xy$. Let $Trade(a, b, 1)(x, w)$ describe the action where Anne trades her Flax card for the Wheat card from Bill (meaning: they hand over their cards to each other in such a way that Cath observes the trade but cannot see the cards). We can then express that after that trade Anne can corner the market in Wheat, by the formula $[Trade(a, b, 1)(x, w)]Corner(a, w)$. The modal operator $[Trade(a, b, 1)(x, w)]$ is interpreted as a state transition that transforms the game state describing the knowledge of the players in deal $wx.wy.xy$ to the new game state describing the knowledge of the players after the trade, when the deal has become $ww.wx.xy$. In other words, a state transition is a binary relation between game states (information states). A game state is not the same as a card deal: every game state is about one card deal, but the same card deal can occur in many game states. The formal counterpart is given in the Appendix. Combinations of modal operators for game actions and for knowledge of players is also conceivable. For example, $[Trade(a, b, 1)(x, y)]K_a K_b y_a$ describes that, after Anne and Bill swapped a Flax and a Rye card, Anne knows that Bill knows that she holds a Rye card.

A new feature in this epistemic action language, that has not been reported before[1], is the 'assignment': assignments are useful to describe that cards change hands: if Anne actually trades the Flax card x^1, fact x^1_a – 'Anne holds card x^1' – becomes false, and if Bill gets that card, fact x^1_b becomes true.[2] For the present purpose, assignments of facts to true or false suffice: $q_n := \top$ and $q_n := \bot$ express that it becomes true and false, respectively, that player n holds card q. For reference purposes we give the semantics of assignment plus an example in the Appendix.

Other action constructs are explained while defining the Pit game actions, now to follow. The main thing to keep in mind here, is that these action descriptions are not 'merely' informal descriptions but that these are truly computable actions strictly and only defined as relational state transitions, or, more in computer science terms, as paths in a large process graph representing all static and dynamic Pit game features for a given set of players and cards. Therefore every statement about the game can effectively be decided to be either true or false. We start with an overview of the abstract descriptions of the five sorts of action in Pit, illustrated by the examples in SixPit. In the explanatory text, the parts between parentheses are the the parameters of the actions that are not public.

- $\mathit{Dealcards}(wx.wy.xy)$
  The cards are dealt to the players. All players pick up their cards. (Anne holds Wheat and Flax. Bill holds Wheat and Rye. Cath holds Flax and Rye.)

- $\mathit{Offer}(a, 1)(w)$
  Anne offers one card for trade. (It is a Wheat card.)

- $\mathit{Trade}(a, b, 1)(w, y)$
  Anne and Bill trade one card. (Anne trades Wheat. Bill trade Flax.)

- $\mathit{Nocorner}(a, b)$
  After trading cards, neither Anne nor Bill has achieved a corner.

- $\mathit{Cornered}(a, w)$
  Anne corners the market in Wheat.

The remainder of this section makes these descriptions semantically precise. The next section, that presents some game theoretical results for Pit, can be read independently from these semantical precisions and only requires understanding of the informal descriptions above.

[1] Work in progress by Hans van Ditmarsch, Wiebe van der Hoek, and Barteld Kooi.

[2] We can even say that the (new) value of the fact 'Bill holds card x^1' becomes the (old) value of 'Anne holds card x^1', in program form: $x^1_b := x^1_a$. But we then still have to make x^1_a false as well.

4.1 Details of game action descriptions

Dealing the cards to the players

$$\begin{array}{lcl} \mathit{Deals} & := & \bigvee_{deal} \delta^{deal} \\ \mathit{Learnhand}_n & := & L_N \bigcup_{hand} L_n?hand_n \\ \mathit{Dealcards} & := & L_N?\mathit{Deals} \ ; \ \mathit{Learnhand}_n \ ; \ \mathit{Learnhand}_{n'} ... \end{array}$$

A paraphrase in slightly more natural language of the description *Dealcards* is: "All players (N) learn (L) that the test (?) on the formula '*Deals*', describing that each player holds a full hand of m cards, succeeds. Then (;) the program '$\mathit{Learnhand}_n$' is executed, describing that all players learn that player n learns his/her hand of cards. Then ... (etc., for each player)" The order of players looking up their cards is irrelevant in the strong sense that bisimilar structures result.

An example in terms of SixPit: The cards are dealt over Anne, Bill and Cath. Anne then picks up her cards and looks at them. (She has a Wheat and a Flax card.) Bill then picks up his cards and looks at them. (He has a Wheat and a Rye card.) Finally Cath picks up her cards and looks at them. (Cath has a Flax and a Rye card.) The non-parenthesized part of each of these four actions is commonly observed by Anne, Bill and Cath.

The formula '*Deals*' is the disjunction of the descriptions δ^{deal} for all possible card deals *deal* of $|N| \cdot m$ cards over the players, where each player gets m cards. For example, the deal where Anne gets cards w^1, x^1, Bill gets cards w^2 and y^1, and Cath gets cards x^2 and y^2 is described by (a conjunction of 18 literals namely)

$$w^1_a \wedge \neg w^2_a \wedge x^1_a \wedge \neg x^2_a \wedge \neg y^1_a \wedge \neg y^2_a \wedge \neg w^1_b \wedge ... \wedge y^2_c$$

Program $\mathit{Learnhand}_n$ is defined as nondeterministic choice between, for all hands of cards, player n learning his hand '*hand*' of cards, where all players observe that. For example, for Anne, who is handed two cards, $\mathit{Learnhand}_n$ states that the players learn that Anne learns which of nine different hands she has been dealt

$$L_{abc}(L_a?ww_a \cup L_a?wx_a \cup L_a?wy_a \cup L_a?xw_a \cup L_a?xx_a \cup ...L_a?yw_a \cup ...)$$

Given the actual card deal $wx.wy.xy$, this action is executable because the alternative $L_a?wx_a$ is executable. We can describe this differently as a deterministic action

$$\mathit{Learnhand}_a(wx)$$

which selects, so to speak, 'only for Anne, while the others remain ignorant' alternative $L_a?wx_a$ before execution. Neither Bill nor Cath know what Anne's hand is after executing $\mathit{Learnhand}_a(wx)$ in deal $wx.wy.xy$. But $\mathit{Learnhand}_a$ can also be executed in $xx.wy.wy$, where Anne's hand is xx, whereas $\mathit{Learnhand}_a(wx)$ cannot. Similarly we also get a deterministic 'card deal' action, for example,

$$\mathit{Dealcards}(wx.wy.xy)$$

that is only executable for a specific deal. This switch from nondeterministic to deterministic is a 'semantic trick' in the action language that is called 'local choice'. We give no details.

Offering cards for trade

$$\mathit{Offer}(n, i) := L_N?\bigvee_{u \in U} u_n^{+i}$$

In other words: everybody learns that player n has at least i cards of the same suit. For example, $L_{abc}?(w_a \vee x_a \vee y_a)$ describes that Anne offers a single card for trade. The cards that are being offered are not made public. The functional version of this nondeterministic action is, for the concrete case of Anne offering Wheat:

$$\mathit{Offer}(a, 1)(w)$$

Again, note that Bill and Cath do not know that Anne offers Wheat. The *Offer* action is of course only informative for larger hands of cards: if there were nine cards altogether, and all players therefore hold three cards, then if Anne is offering two cards for trade, both Bill and Cath *know* that her hand cannot be wxy. This resulting knowledge is now formally captured as (part of) a postcondition of the *Trade* action.

Trading cards Suppose that Anne trades a Wheat card with a Rye card from Bill. More precision is actually needed for specification of this action. Assume that Anne held w^1 and Bill held y^2. Anne pushes card w^1 facedown over to Bill, and simultaneously Bill pushes card y^2 over to Anne. At this stage, the cards can be said to have changed 'hands' in the sense of ownership: the value of the corresponding facts need to be reassigned. Anne then picks up card y^2, after which Bill picks up card w^1. By picking up cards, the player learns the suit of those cards, or, in terms that we can model "learns the suit of the cards in his possession of which he doesn't know the suit yet". Again, just as when the cards were dealt, the order of Anne and Bill picking up their new card is irrelevant: if Bill picks it up first and then Anne, the same (i.e., bisimilar) information state will result. In general, we get the following description; for simplicity, we stick to the case of a *single* card being traded, the action where player n and player n' trade more than one card (of the same suit) is similarly defined as $\mathit{Trade}(n, n', i)$:

$$\begin{array}{lcl}
\mathit{Swap}(n, n', 1)(q, q') & := & q_n := \bot \ ; \ q_{n'} := \top \ ; \ {q'}_{n'} := \bot \ ; \ {q'}_n := \top \\
\mathit{Swap}(n, n', 1)(u, u') & := & \bigcup_{q \in u, q' \in u', q \neq q'} \mathit{Swap}(n, n', 1)(q, q') \\
\mathit{Swap}(n, n', 1) & := & \bigcup_{u \in U} \mathit{Swap}(n, n', 1)(u, u') \\
\mathit{Learncard}_n & := & L_N \bigcup_{u \in U; i \leq m} L_n?(u_n^{+i} \wedge \neg K_n u_n^{+i}) \\
\mathit{Trade}(n, n', 1) & := & L_N \mathit{Swap}(n, n', 1) \ ; \ \mathit{Learncard}_n \ ; \ \mathit{Learncard}_{n'}
\end{array}$$

The order of the four assignments in $\mathit{Swap}(n, n', 1)(q, q')$ is irrelevant. But it is highly relevant that keeping the assignments together means that that the cards are traded simultaneously and not one after the other: if a player gives a card to another player and then that other player returns a card, *that may have been the card that other player had just been given.* This is ruled out by our specification.

The 'test formula' (the formula preceded with '?') in the action $Learncard_n$ indeed describes that for some suit u of cards, player n holds one more card of that suit (namely i cards) than he currently knows. That is the card that he has just been handed.

The functional version of the nondeterministic action $Learncard_n$ is $Learncard_n(u)$ and the corresponding functional version of two agents trading a single card is $Trade(n, n', 1)(u, u')$. For example, the action of Anne trading a Wheat card with a Rye card from Bill is described by $Trade(a, b)(w, y)$.

As the assignment action is a recent addition to the dynamic language that we presume in these modellings, we give its formal details in the Appendix to this contribution. This includes an example illustrating its meaning.

Not cornering the market

$$Nocorner(n, n') := L_N?(\neg Corner(n) \wedge \neg Corner(n'))$$

In other words: everybody learns that the two players that just traded cannot corner the market, as neither declared a corner. We remind the reader that, for example, $Corner(a)$ in the six card case is defined as $Corner(a, w) \vee Corner(a, x) \vee Corner(a, y)$ which means $ww_a \vee xx_a \vee yy_a$: Anne corners the market if she holds two Wheat cards, or two Flax cards, or two Rye cards.

This is an implicit action. Its absence appears from players starting to make next offers, so to speak. For synchronization purposes we have to distinguish this action as 'one more tick of the clock': a proper action that is being observed by all players, before they continue to offer cards for trade.

Cornering the market

$$Cornered(n, u) := L_N?Corner(n, u)$$

This description will by now be obvious. Note that the game requires the player to declare the commodity of his corner. The suit is therefore public. This is indeed how we have modelled it.

This concludes the detailed description of game actions in Pit. For an example, in the Appendix we show the information states that result from executing these actions in SixPit, if the card deal is $wx.wy.xy$ and the cards have just been dealt.

5 Game theory of Pit

To our knowledge, no results are known on the game theory of Pit. Very few investigations have been made anyway into the game theory of knowledge games

[2]. We make a small exploration into this matter. Note that we have assumed, for simplification of matters, that all cards have the same value, and that chance, and not choice, determines what players are allowed to trade with each other. The simplest system that is not trivial is the Pit game for three players and six cards, i.e., two cards per player. This is the SixPit game already described above. We will restrict our investigations in this Section to SixPit. The players are, as before, a, b, and c, and the cards are w, w, x, x, y, y. For our current purposes there is no need to give different cards of the same suit different names, as in the previous section. We start by a random distribution of these six cards over the three players a, b, c, where they all get two cards.

An example of the only type of deal that does not immediately end in a corner is $wx.wy.xy$. By this representation, we mean that Anne holds $\{w, x\}$, Bill holds $\{w, y\}$, and Cath holds $\{x, y\}$. Given deal $wx.wy.xy$, all players seem to be 'equally well' informed. Their actual information is of course different. For example, Anne does not know that Bill's hand is wy whereas Bill knows that his hand is wy. But modulo a permutation of suits and agents, their information is the same: Bill does not know that Anne's hand is wx. Because of that, none has an advantage over the others after the cards have been dealt. Also, the players cannot make an intelligent choice between the two cards in their hand. They cannot do better than randomly choosing a card for trade! Well, almost: they can choose between not offering for trade or offering one card for trade. (Two cards – that have to be of the same suit! – is out, because that would have been a corner already.) 'Not trading' is irrational: if you do not offer cards for trade, you do not have a chance to win, even though the penalty for losing is just as high as when you trade and do not corner the market, or when you offer for trade but are not selected; whereas if you offer cards for trade, you have a non-zero probability of getting selected for trade and if you are a non-zero probability of winning in that move.

As before, a player does not disclose which card he offers for trade, and must determine the card before and not after the offer is honoured. Given the offers, it is determined randomly which offers are matched, 'after which' (instantly) that trade takes place.

Suppose that given the card deal $wx.wy.xy$ at the outset of the game, Anne and Bill are chosen to trade the card they offer. Then there are four possibilities[3]:

Current deal	*Game action*	*Resulting deal*
$wx.wy.xy$	$Trade(a, b, 1)(w, w)$	$wx.wy.xy$
$wx.wy.xy$	$Trade(a, b, 1)(w, y)$	$xy.ww.xy$
$wx.wy.xy$	$Trade(a, b, 1)(x, w)$	$ww.xy.xy$
$wx.wy.xy$	$Trade(a, b, 1)(x, y)$	$wy.wx.xy$

In the third case Anne will declare a corner in Wheat and wins, and in the second case Bill declares a corner in Wheat and wins. In the other cases a

[3] In the Appendix the formal counterparts to the game states that correspond to the current and resulting card deals are visualized.

further move has to be made, possibly ad infinitum. They appear rather boring, in particular the first case: we already *had* deal $wx.wy.xy$, and now we have it again. Appearences can be deceptive. Relevant is, what the game state was and has become, not merely what the card deal was and has become. Game states are the pointed Kripke models (relational structures) from the previous Section: they include what players know about each other. And in the game state for deal $wx.wy.xy$ resulting from game action $\mathit{Trade}(a, b, 1)(w, w)$, both Anne and Bill, but not Cath, happen to know what the card deal is. (And this was not the case in the game state for the same card deal before that action.) We can establish this by reasoning about the knowledge that the players have about each others' cards. Anne knows that Bill now has a Wheat card, because she just gave him her own. But Anne can now also deduce Bill's other card: it cannot also be Wheat, because she received Wheat and if Bill's other card had been Wheat he would immediately have declared a corner in Wheat and not have offered a card for trade. But it cannot be Flax either, because then Cath would have had two Rye cards initially and immediately have declared a corner in Rye. And she didn't. So Bill's other card must be Rye. But then Cath holds the two remaining cards: Flax and Rye. In other words: Anne knows the deal of cards. Bill can reason as Anne, and therefore also knows the deal of cards. Cath has not gained factual knowledge from Anne and Bill trading. But, e.g., she now knows that Anne and Bill know the deal of cards.

We can now make several meaningful observations:

First, Anne can now distinguish between 'the card in her hand that she shares with Bill, with whom she just traded', namely her Wheat card, and 'the card in her hand that she shares with Cath, with whom she did not just trade', namely her Flax card. Therefore, in the next move of the game, she may have an individual preference to choose to offer for trade the one or the other card, or even a strategic choice based on the similar preferences of the other player that has just traded, Bill. Cath cannot make an intelligent choice between one of her cards to offer for trade, because she was not selected for the previous trade, and because her knowledge of Anne and Bill is symmetric with respect to her cards.

Second, the justification for these distinctions are observations about *knowledge* that the players have about each other, not merely about *facts*, i.e., not merely about their hands of cards. In the case of SixPit, the argument was fairly simple. We could do it 'by hand' so to speak. But in the general case this may require checking possibly complex knowledge properties of the players. Those properties are entailed by the logical description of the game actions and game states in the previous section. Therefore, this logical description is indispensible in order to define the game, even up to - as we will see - the computation of players' individual preferences. This is a typicality of such 'knowledge games': generally in game theory, all of the game states, actions, payoffs, and induced preferences, are a *given*, a parameter that one starts out with. Not so for knowledge games: both game states and game actions have a structured, computable description, such that preferences are also computable given a payoff corresponding to some game state feature. We will largely bypass the epistemic details of game states, but in the Appendix we have visualized the four actions discussed here as transitions between pointed relational structures, including part of the further development of that game that we will analyze after these observations.

Third, this happens to be not just an interesting game state on level one of the game tree, but, modulo the agent and suit symmetries also mentioned above, the *single* game state ever occurring in any further development of the game. Whatever the next game move is, either a corner results (and, unlike in the initial deal of cards, always for *one* player only) or a similar information state will result where the two players that were allowed to trade now know the deal of cards, but the player who did not trade does not know it (and therefore may even have 'forgotten' his knowledge of the deal before the trade took place).

This can be shown as follows: Given was the distribution $wx.wy.xy$. In the next move, either a and b, or a and c, or b and c trade a card. *If* a and b trade again, then the same argument applies as above. *If* a and c trade instead, b no longer knows the cards of a and c, because he does not know which of their two cards either of them has chosen to trade. On the other hand, a and c now learn the entire deal of cards, as above. And if b and c trade instead, then that case is analogous to a and c trading. So whatever behaviour is optimal for this move in the game, will be similarly optimal for any future game state that does not end the game.

Therefore, for each player, there are only two 'game actions' in the sense of game theory.[4] These are:

- shared_n
  "if player n traded in the previous round, then he offers the card for trade that he knows to share with the player whom he traded with, and otherwise he chooses his card randomly."

- $\mathit{distinct}_n$
  "if player n traded in the previous round, then he offers the card for trade that he knows to share with the player whom he did not trade with, and otherwise he chooses his card randomly."

The individual preferences of the players are according to the expected payoffs of these actions. This then defines the game and the optimal strategies for the players [6]. We present a full analysis for a further simplification of SixPit, namely that consisting of one offer/trade action only, and an outline of the equilibria for SixPit, that may require an arbitrarily large number of repeats of these actions.

[4] Here we see one of the discrepancies between the 'logical' and the 'economics' point of view: for example, one of the executions of a profile – pair consisting of two individual 'economics' game actions – $(\mathit{shared}_a, \mathit{distinct}_b, \mathit{distinct}_c)$ in the game state for deal $wx.wy.xy$ resulting from Anne and Bert have traded a Wheat card, is the program – sequence of three 'logic' game actions – $\mathit{Offer}(a)(w)$; $\mathit{Offer}(b, y)$; $\mathit{Trade}(a, b)(w, y)$ (namely the execution where a and b were selected for trade).

There is a difference between a player offering a card for trade, and a player choosing an action. The first is an execution instance of the second. Assume the game state for deal $wx.wy.xy$ resulting from (epistemic) action $\mathit{Trade}(a, b, 1)(w, w)$ in the (different) game state for (the same) deal $wx.wy.xy$. Suppose that player a now chooses shared_a, that player b chooses $\mathit{distinct}_b$, and that player c chooses shared_c. This means that player a offers card w for trade, player b offers card y for trade, and player c randomly chooses on of her cards for trade, suppose the result is that she offers card y for trade. If a and b are now selected for trade, the result of $\mathit{Trade}(a, b, 1)(w, y)$ is the deal $xy.ww.xy$: b corners the market in w and wins the game. If, instead a and c had been selected for trade, the result of $\mathit{Trade}(a, c, 1)(w, x)$ is the deal $xy.wy.wx$. Nobody wins in this round and (at least) another move has to be made to finish the game. The three matrices below present a systematic overview of these payoffs. We emphasize that these are *not* game matrices, as w, x, and y do not meaningfully represent game actions.

$a \backslash b$	w	y
w	$-$	$(-1, 2, -1)$
x	$(2, -1, -1)$	$-$

$a \backslash c$	x	y
w	$(2, -1, -1)$	$-$
x	$-$	$(-1, -1, 2)$

$b \backslash c$	x	y
w	$-$	$(-1, 2, -1)$
y	$(-1, -1, 2)$	$-$

For example, the result of action $\mathit{Trade}(a, b, 1)(w, y)$, after which b wins and therefore a and c lose, is found in the leftmost matrix in column 2, row 1. Payoff $(-1, 2, -1)$ means that player a loses 1, player b gains 2, and player c loses 1. Note that we have defined the combined payoff three-player zero-sum. This makes sense, as the only objective of the game is to be the first to corner the market, resulting in the others not doing that. Payoff '$-$' means that more moves are needed to finish the game. Once more, we simplify the game: we assume that in those case the payoff is $(0, 0, 0)$. After we have described the equilibria for that game, we continue our analysis of the full game. For example, the first of the three matrices above has become:

$a \backslash b$	w	y
w	$(0, 0, 0)$	$(-1, 2, -1)$
x	$(2, -1, -1)$	$(0, 0, 0)$

We can now compute the expected payoff of the game for each player, given that they each choose between their two possible strategies with some probability. As the previous trade was between a and b, c's behaviour can be assumed to be random. Therefore, we can model this simplified SixPit move as a two-player game namely between a and b only. Let p_n be the probability with which player

n chooses *shared*$_n$ such that $(1-p_n)$ is the probability with which that player chooses action *distinct*$_n$. The expected payoff $E_a(p_a,p_b)$ for player a is:

$$\begin{array}{rcl} E_a(p_a,p_b) & = & (\frac{1}{3}\cdot p_a\cdot(1-p_b)\cdot -1)+(\frac{1}{3}\cdot(1-p_a)\cdot p_b\cdot 2)+ \\ & & (\frac{1}{3}\cdot p_a\cdot \frac{1}{2}\cdot 2)+(\frac{1}{3}\cdot(1-p_a)\cdot\frac{1}{2}\cdot -1)+ \\ & & (\frac{1}{3}\cdot p_b\cdot\frac{1}{2}\cdot -1)+(\frac{1}{3}\cdot(1-p_b)\cdot\frac{1}{2}\cdot -1) \\ & = & \frac{2}{3}p_b+\frac{1}{6}p_a-\frac{1}{3}p_ap_b-\frac{1}{3} \end{array}$$

On symmetry grounds the expected payoff for a and b are of course analogous, and the expected payoff for c is 'whatever is needed to add up to 0', as obviously any combination of strategies is also zero-sum for a, b, and c together. We therefore get:

$$\begin{array}{rcl} E_b(p_a,p_b) & = & \frac{2}{3}p_a+\frac{1}{6}p_b-\frac{1}{3}p_ap_b-\frac{1}{3} \\ E_c(p_a,p_b) & = & -\frac{5}{6}p_a-\frac{5}{6}p_b+\frac{2}{3}p_ap_b+\frac{2}{3} \end{array}$$

We have now finally reached the level of defining a standard game. The resulting game matrix for players a and b takes the following shape. For example, the combined payoff when a plays *shared*$_a$ and b plays *distinct*$_b$ corresponds to $(E_a(1,0),E_b(1,0))$ above, etc. More in line with how we have chosen the probabilities, *distinct*$_a$ comes first, and *shared*$_a$ comes second, and similarly for b:

$a\backslash b$	*distinct*$_b$	*shared*$_b$
distinct$_a$	$(-\frac{1}{3},-\frac{1}{3})$	$(\frac{1}{3},-\frac{1}{6})$
shared$_a$	$(-\frac{1}{6},\frac{1}{3})$	$(\frac{1}{6},\frac{1}{6})$

This game has two equilibria (*distinct*$_a$, *shared*$_b$), and (*shared*$_a$, *distinct*$_b$), and a mixed equilibrium for $p_a=p_b=\frac{1}{2}$, i.e., a playing $\frac{1}{2}\cdot$ *distinct*$_a$ $+\frac{1}{2}\cdot$ *shared*$_a$ and b playing $\frac{1}{2}\cdot$ *distinct*$_b$ $+\frac{1}{2}\cdot$ *shared*$_b$. In the third equilibrium the combined payoff is $(0,0)$. A peculiar property is associated with that third equilibrium: when a plays a random card, b cannot affect his own expected payoff but only a's expected payoff, and vice versa. For example, for $p_a=\frac{1}{2}$, we get that $E_a(\frac{1}{2},p_b)=\frac{1}{2}p_b-\frac{1}{4}$ and $E_b(\frac{1}{2},p_b)=0$. In other words: when a doesn't think at all, b cannot take advantage of that.

It is further worthwhile to observe that the game matrix is of a so-called 'chicken-like' game, where playing *shared*$_n$ may be seen as the cooperating strategy and playing *distinct*$_n$ as the defecting strategy. As in 'chicken', also here $(1,0)$ and $(0,1)$ are equilibria.

In the context of the simplified SixPit 'knowledge game' we can interpret the unstable profile (*shared*$_a$, *shared*$_b$) as follows: *if* Anne and Bill form a coalition, they can outwit Cath and each increase their expected gain from 0 to $\frac{1}{6}$. It is unstable, as for either of them it is profitable to 'break up the coalition' and act in their private interest: if the other doesn't they further increase their expected gain to $\frac{1}{3}$. Unfortunately, if they both do that, they both lose $\frac{1}{3}$ instead, and Cath is the laughing bystander who then gains.

Towards the full SixPit game We now present the outline of a generalization to the SixPit game wherein the 'game' above just describes one move, namely a typical second move of the game. If we assume that players choose a strategy before they start playing the game (and that, therefore, they do not reward or punish each other for their behaviour during the game), we arrive at an appealing though still rather complex generalization. The payoff

$$(0,0,0)$$

in the 'one-move game' matrix can now be replaced with expected payoffs that we write, for that specific continuation, as

$$(E_a^{ab}(p_a,p_b,p_c), E_b^{ab}(p_a,p_b,p_c), E_c^{ab}(p_a,p_b,p_c))$$

where, for example, $E_a^{ab}(p_a,p_b,p_c)$ is the expected gain of player a in the remainder of the game, given that a traded with b in the current move and that the game is not over yet, and given that a (always) plays *shared*$_a$ with probability p_a and b plays *shared*$_b$ with probability p_b and c plays *shared*$_c$ with probability p_c. We therefore get the following 'game-like matrices' (write E_a^{ab} for $E_a^{ab}(p_a,p_b,p_c)$, etc.) to help us compute the expected payoff E^{ab} of the players given that a and b have just traded:

$a\backslash b$	w	y
w	$(E_a^{ab}, E_b^{ab}, E_c^{ab})$	$(-1,2,-1)$
x	$(2,-1,-1)$	$(E_a^{ab}, E_b^{ab}, E_c^{ab})$

$a\backslash c$	x	y
w	$(2,-1,-1)$	$(E_a^{ac}, E_b^{ac}, E_c^{ac})$
x	$(E_a^{ac}, E_b^{ac}, E_c^{ac})$	$(-1,-1,2)$

$b\backslash c$	x	y
w	$(E_a^{bc}, E_b^{bc}, E_c^{bc})$	$(-1,2,-1)$
y	$(-1,-1,2)$	$(E_a^{bc}, E_b^{bc}, E_c^{bc})$

From this, we can, for example, compute E_a^{ab} as follows:

$$\begin{aligned}
E_a^{ab}(p_a,p_b,p_c) \quad = \quad & \tfrac{1}{3}\cdot p_a\cdot p_b\cdot E_a^{ab}+\\
& \tfrac{1}{3}\cdot p_a\cdot(1-p_b)\cdot -1+\\
& \tfrac{1}{3}\cdot(1-p_a)\cdot p_b\cdot 2+\\
& \tfrac{1}{3}\cdot(1-p_a)\cdot(1-p_b)\cdot E_a^{ab}+\\
& \tfrac{1}{3}\cdot p_a\cdot\tfrac{1}{2}\cdot 2+\\
& \tfrac{1}{3}\cdot p_a\cdot\tfrac{1}{2}\cdot E_a^{ac}+\\
& \tfrac{1}{3}\cdot(1-p_a)\cdot\tfrac{1}{2}\cdot E_a^{ac}+\\
& \tfrac{1}{3}\cdot(1-p_a)\cdot\tfrac{1}{2}\cdot -1+\\
& \tfrac{1}{3}\cdot p_b\cdot\tfrac{1}{2}\cdot E_a^{bc}+\\
& \tfrac{1}{3}\cdot p_b\cdot\tfrac{1}{2}\cdot -1+\\
& \tfrac{1}{3}\cdot(1-p_b)\cdot\tfrac{1}{2}\cdot -1+\\
& \tfrac{1}{3}\cdot(1-p_b)\cdot\tfrac{1}{2}\cdot E_a^{bc}
\end{aligned}$$

Obviously p_c does not *yet* occur in the equation, but this is uncovered by both E_a^{ac} and E_a^{bc}, for example, the equation for $E_a^{ac}(p_a,p_b,p_c)$ contains a (first) part

$\frac{1}{3}p_a p_c E_a^{ac}(p_a, p_b, p_c)$ and another (one of twelve) part(s) $\frac{1}{6}p_c E^{bc}(p_a, p_b, p_c)$. This results in a fairly complex set of nine recurrent equations in three variables, for which one can in principle find a solution and possibly equilibria. We are still investigating what the optimal strategies for this full SixPit game are. The general pattern of these investigations is clear:

In the SixPit game there are after the first move only *three* different epistemic states, depending on the two players that just traded. Therefore there are *three* expected payoff functions E^{ab}, E^{ac}, and E^{bc}. In the general case of a Pit game for an arbitrary number of players and cards, we can compute for each distinct epistemic state of the game, as modelled by the logical formalization presented in the previous section, the expected payoffs for each player in a way very similar to the above procedure.

6 Conclusions and further research

We have presented an overview of game states and game actions occurring in a simplified version of the Pit game. Thus we have grounded the investigation of this and similar simplifications into the mainstream of formal logical semantics. We have presented some minor results of game theoretical relevance for Pit, in particular for the SixPit game for three players and six cards only. This revealed the 'game actions' that are the game theoretical counterparts of the logical 'game actions' that were described separately. It also clearly demonstrated that the logical specification is indispensable for the game theoretical investigations. We described equilibria of the SixPit game.

Further research will involve different abstractions of the Pit game, for example allowing for simultaneous trading between different couples of players. This presents no logical difficulties, but is mainly interesting for game theory. Such explorations might link our results to those from [7]. We are also implementing simple Pit games in an epistemic model checker. This would allow to check automatically what players know in a given game state. We further intend to generalize the current game theoretical results. Finally, Johan Lövdahl has implemented some Pit game plays for demonstration purposes for a three player and twelve cards case. This can be viewed on webpage `http://www.ida.liu.se/~jolov/pit/`.

References

[1] A. Baltag, L.S. Moss, and S. Solecki. The logic of public announcements, common knowledge and private suspicions. Manuscript, originally presented at TARK 98, 2003.

[2] S. Druiven. Knowledge development in games of imperfect information. Master's thesis, University of Groningen and University of Maastricht, Groningen / Maastricht, the Netherlands, 2002.

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

[4] J.D. Gerbrandy. *Bisimulations on Planet Kripke*. PhD thesis, University of Amsterdam, 1999. ILLC Dissertation Series DS-1999-01.

[5] C.A. Holt. Trading in a pit market. *Journal of Economic Perspectives*, 10(1):193–203, 1996.

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

[7] M. Purvis, M. Nowostawski, S. Cranefield, and M. Oliveira. Multi-agent interaction technology for peer-to-peer computing in electronic trading environments. In G. Moro, C. Sartori, and M. Singh, editors, *Second International Workshop on Agents and Peer-to-Peer Computing*, pages 103–114, Melbourne, 2003.

[8] H.P. van Ditmarsch. *Knowledge games*. PhD thesis, University of Groningen, 2000. ILLC Dissertation Series DS-2000-06.

[9] H.P. van Ditmarsch. Descriptions of game actions. *Journal of Logic, Language and Information*, 11:349–365, 2002.

[10] H.P. van Ditmarsch, W. van der Hoek, and B.P. Kooi. Concurrent dynamic epistemic logic. In V.F. Hendricks, K.F. Jorgensen, and S.A. Pedersen, editors, *Knowledge Contributors*, pages 45–82, Dordrecht, 2003. Kluwer Academic Publishers. Synthese Library Volume 322.

[11] H.P. van Ditmarsch, W. van der Hoek, and B.P. Kooi. Descriptions of game states. In G. Mints and R. Muskens, editors, *Logic, Games, and Constructive Sets*, pages 43–58, Stanford, 2003. CSLI Publications. CSLI Lecture Notes No. 161.

Appendix: dynamic epistemics with assignment

Given are a set of atoms P and agents N, a two-typed dynamic epistemic logical language $\mathcal{L}_N(P)$ with both formulas φ and actions α, and a class of structures for atoms P and agents N called epistemic states. An epistemic state $(M, s) = (\langle S, \sim, V\rangle, s)$ consists of a domain of abstract objects, with for each agent an equivalence relation $\sim_n$ on the domain, and for each atom p a subset V_p of the domain, and a designated object that is the actual state of affairs. Epistemic – and other - actions are defined by state transitions for dynamic modal operators as follows:

$$(M, s) \models [\alpha]\varphi \quad \text{iff} \quad (M', s') \models \varphi \text{ for all } (M', s') \text{ such that } (M, s)[\![\alpha]\!](M', s')$$

The assignment action ('program') is one particular example of such an action. It is defined as follows:

$$(M, s)[\![p := \varphi]\!](M', s') \quad \text{iff} \quad M' = \langle S, \emptyset, V'\rangle \text{ and } s' = s$$

where V' is as V except that $V_p := S_\varphi$, such that $S_\varphi = \{s \in S \mid M, s \models \varphi\}$.

For an example, in Figure 1 we visualize the action where three players Anne, Bill and Cath (a, b, c) each hold *one* card namely Wheat, Flax, and Rye (w, x, y) and where Anne and Bill swap their cards and first Anne looks at her card and then Bill looks at his card. The deal is represented by $w.x.y$ (as before), and

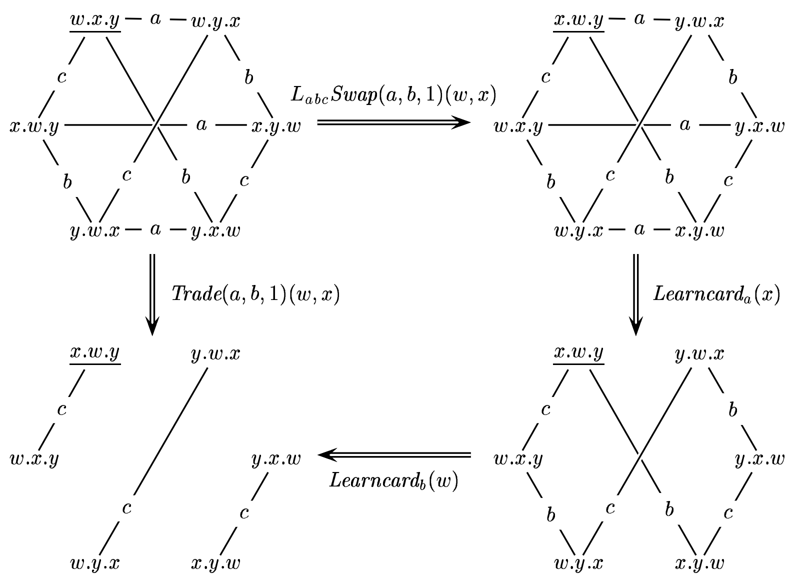

Figure 1: First, Anne and Bill swap their card. Then, Anne looks at her card. Then, Bill looks at his card. In the resulting state, Anne and Bill know the deal of cards, but Cath does not.

we have that, initially, for Anne deal $w.x.y$ cannot be distinguished from deal $w.y.x$ (where Bill holds Rye and Cath holds Flax), etc. After the cards have been swapped, Anne knows that Bill holds her former card, Wheat, but does not know her own card yet. For that, she has to look at it first. After Anne and Bill have both looked, they know the card deal, but Cath still cannot distinguish between $w.x.y$ and $x.w.y$: she does not know which of Anne of Bill holds Wheat, etc.

For the SixPit game, the trade action gives of course quite similar results, but with different numbers of distinct states in the models. In that case, there are 21 different card deals. It so happens that all but six of those involve a corner. Somewhat by coincidence we can visualize a game state in SixPit after the cards have been dealt and possible corners have been declared in a very similar 'hexagonal' figure. The swap action then leads to a model with 24 distinct states, because for each deal, there are four ways for two players to swap a card. Subsequent possible 'Cornering' actions reduce the model again. If we combine these actions, we get an appealing result again: Figure 2 that pictures what is known when, given the deal $wx.wy.xy$, a and b are chosen for trade. Any subsequent development in SixPit from either of the non-terminal nodes copies this pattern, the game defined in Section 5 corresponds to a single move from node $\underline{wx.wy.xy}$—c—$wy.wx.xy$ in Figure 2.

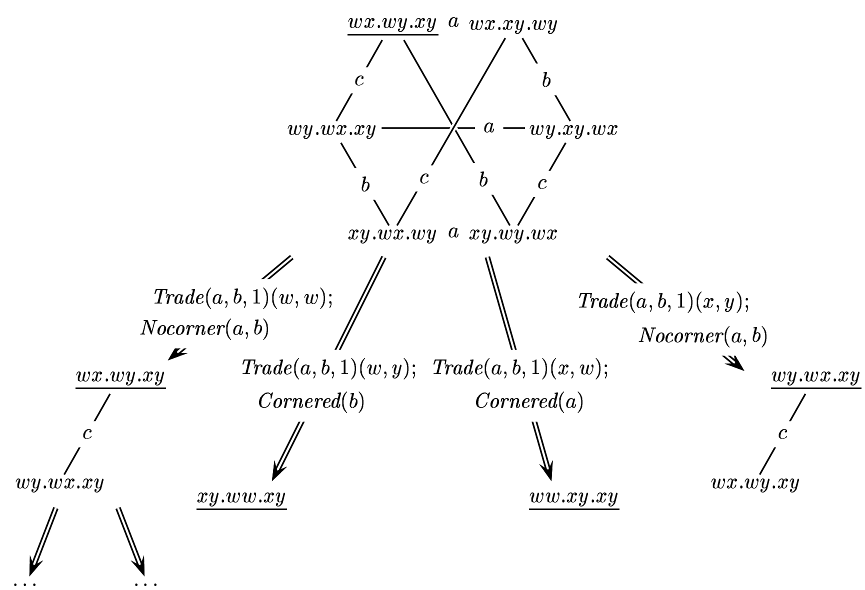

Figure 2: Part of the game tree for SixPit