Timeability of Extensive-Form Games

Sune K. Jakobsen
Queen Mary, University of London
Mile End Road
London E1 4NS
s.k.jakobsen@qmul.ac.uk

Troels B. Sørensen
IT-University of Copenhagen
Rued Langgaards Vej 7
DK-2300 København S
trbj@itu.dk

Vincent Conitzer
Duke University
Box 90129
Durham, NC 27708, USA
conitzer@cs.duke.edu

ABSTRACT

Extensive-form games constitute the standard representation scheme for games with a temporal component. But do all extensive-form games correspond to protocols that we can implement in the real world? We often rule out games with *imperfect recall*, which prescribe that an agent forget something that she knew before. In this paper, we show that even some games with perfect recall can be problematic to implement. Specifically, we show that if the agents have a sense of time passing (say, access to a clock), then some extensive-form games can no longer be implemented; no matter how we attempt to time the game, some information will leak to the agents that they are not supposed to have. We say such a game is not *exactly timeable*. We provide easy-to-check necessary and sufficient conditions for a game to be exactly timeable. Most of the technical depth of the paper concerns how to *approximately* time games, which we show can always be done, though it may require large amounts of time. Specifically, we show that some games require time proportional to the power tower of height proportional to the number of players, which in practice would make them untimeable. We hope to convince the reader that timeability should be a standard assumption, just as perfect recall is today. Besides the conceptual contribution to game theory, we show that timeability has implications for onion routing protocols.

Categories and Subject Descriptors

I.2.11 [**Distributed Artificial Intelligence**]: Multiagent Systems; J.4 [**Computer Applications**]: Social and Behavioral Sciences–Economics

General Terms

Economics, Theory

Keywords

Computational Game Theory; Equilibrium Computation

1. INTRODUCTION

The *extensive form* is a powerful representation scheme for games. It allows one to naturally specify how the game unfolds over time, and what each player knows at each point of action. This allows one to model, for example, card games such as poker, but also real-world strategic situations with similar aspects.

Besides asking whether all strategic situations one might encounter in the real world can be modelled as extensive-form games, one may also ask whether all extensive-form games correspond to something one might encounter in the real world. This question is important for several reasons. One is that if the answer is "no", then there should be some well-motivated restricted subclasses of extensive-form games that may be more tractable from the perspective of algorithmic and other theoretical analysis. Another is that if we are interested in designing a protocol, extensive-form games give us a natural language in which to express the protocol—but this language may lead us astray if some of its games are not actually implementable in the real world.

Games of *imperfect recall*, in which an agent sometimes forgets something she knew before, constitute a natural example of games that may be difficult to implement in the real world.[1] Indeed, restricting attention to perfect recall is often useful for algorithmic and other theoretical purposes. From a theoretical perspective, perfect recall is required [12] for behavioral strategies to be as expressive as mixed strategies. Perfect recall also allows for the use of the sequence form [17], which allows linear optimization techniques to be used for computing equilibria of two-person extensive-form games [19]. The sequence form can also be used to compute equilibrium refinements [14, 13], again requiring perfect recall. Without perfect recall, otherwise simple single agent decision problems become complicated [16, 1, 2], and even the existence of equilibria in behavior strategies becomes NP-hard to decide [5]. Imperfect recall has proven useful for computing approximate minimax strategies for poker [21],

---

[1]Computer poker provides some amusing anecdotes in this regard. When comparing two poker-playing bots by letting them play a sequence of hands, one way to reduce the role of luck and thereby improve statistical significance is to wipe clean the bots' memory and let them play the same sequence of hands again, but with the bots' roles in the hands reversed. This is not feasible for *human* players, of course. Because of this, events pitting computers against humans have generally pitted a *pair* of players against one copy of the bot each, in separate rooms. In this setup, each human-computer pair receives the same hands, though the bot's role in one room is the human's role in the other.

even though the agent following the strategy does have perfect recall when playing the game.

We believe that many researchers are under the impression that, given any finite extensive-form game of perfect recall, one could in principle have agents play that game in the real world, with the actions of the game unfolding in the order suggested by the extensive form. In this paper, we prove that this is not so, at least if agents have a sense of *time*. If the players have a sense of time, we show that some games cannot be implemented in actual time in a way that respects the information sets[2] of the extensive form. The games that can be implemented in time are exactly those that have *chronologically ordered information sets*, as defined in a set of lecture notes by Weibull [23, page 91]. Weibull argues that games with this property constitute the natural domain of *sequential equilibria* [10]. The concept of sequential equilibrium is arguably the most used equilibrium refinement for extensive-form games with imperfect information. Kreps and Ramey [9] provided an example where the unique sequential equilibrium requires some level of cognitive dissonance from the players [22], forcing a player to best-respond to strategies that are not consistent with her beliefs. However, examples of this type only work because they have no ordering of the information sets, which is Weibull's point in restricting attention to games with chronologically ordered information sets. In this paper, we argue something stronger: we argue that extensive-form games without this property cannot model any real world strategic situation, since the information structure of the model cannot be enforced.

We emphasize that our paper is not intended as a criticism of extensive-form games. Rather, the goal is to point out a natural restriction – timeability – that is needed to ensure that the game can be implemented as intended in practice. Again, perfect recall is a restriction that is similar in nature. Restricting attention to those games that have perfect recall has been useful for many purposes, and the notion has also been useful to understand why certain games have odd features—namely, they have imperfect recall. We suspect the notion of timeability can be used similarly. At least one paper already implicitly assumes that all games are timeable [11], while another paper would have been much simpler if it had assumed timeability [20]. We hope that more applications of timeability will be found, and we encourage game theorists (algorithmic or otherwise) to, in contexts where they consider the restriction of perfect recall, consider that of timeability as well.

One place where the analogy between timeability and perfect recall perhaps breaks down is that we have shown that games that are not exactly timeable can nevertheless be approximately timed, in some cases even in a reasonable amount of time. It is not clear whether an analogous notion of approximately perfect recall could be given.

Most of our technical work concerns whether games that do not have an exact timing can nevertheless be approximately timed, and if so, how much time is required to do so. This latter contribution may have important ramifications for the design of protocols that run a risk of leaking information to participants based on the times at which they are requested to take action. While we show that all games are at least approximately timeable, we also show that some games require so much time that in practice they are untimeable.

1.1 Motivating example

Consider the following simple 2-player extensive-form game (Figure 1(a)). In it, first a coin is tossed that determines which player goes first. Then, each player, in turn, is asked to guess whether she has gone first. If the player is correct, she is paid 1 (and otherwise 0). The information sets of the game suggest that a player cannot at all distinguish the situation where she goes first from the one where she goes second, and thus, she gets expected utility 1/2 no matter her strategy.

However, now consider implementing this game in practice. Assume that the game starts at time 0. Clearly, if we toss the coin at time 0, ask one player to bet at time 1, and the other at time 2, a time-aware player will know exactly whether she is being asked first or second (assuming the timing protocol is common knowledge), and will act accordingly. This implementation blatantly violates the intended information structure of the extensive-form representation of the game; indeed, it results in an entirely different game (one that is much more beneficial to the players!). We say that this protocol is not an *exact timing* of the game in Figure 1(a).

Of course, the general protocol of taking one action per time unit is a perfectly fine timing of many games, including games where every action is public (as in, say, Texas Hold'em poker). Also, there are games where taking one action per time unit fails to exactly time the game, but nevertheless an exact timing is available. For example, consider the modified game in Figure 1(b), where player 1 only plays if the coin comes up Heads, and if so plays first. This game can be timed by letting player 1 play at time 1 and player 2 at time 2, even if player 1 does not go first.

But what about the game in Figure 1(a)? Can it not be timed at all? We will pose the constraint that there must be at least one time unit between successive actions in the extensive form. Without this constraint, we could take the normal form of the game and let players play it by declaring their entire strategy at once—but this scheme violates the natural interpretation of the extensive form, and would allow us to play games of imperfect recall just as well. (One may argue that we should just let the players play in parallel after the coin flip in the game in Figure 1(a)—however, a simple modification of the game where the second player is only offered a bet if the first player guessed correctly (Figure 1(c)) would disallow this move.) It is easy to see that no *deterministic* timing will suffice. This is because every node within an information set would have to have the same time associated with it; but then, the left-hand side of the tree requires that player 1's information set has a time strictly before that of player 2, but the right-hand side implies the opposite.

For games where deterministic timing cannot be done, one might turn to randomized timing when trying to implement the game. However, if the time at which a node is played is to reveal *no information whatsoever* about which node in the information set has been reached, then the *distribution* over times at which it is played must be identical for each

---

[2]Recall that for extensive-form games, the information available to the players is represented using *information sets*. Two nodes in a game tree belong to the same information set if they belong to the same player, and the player has the same information at those two nodes.
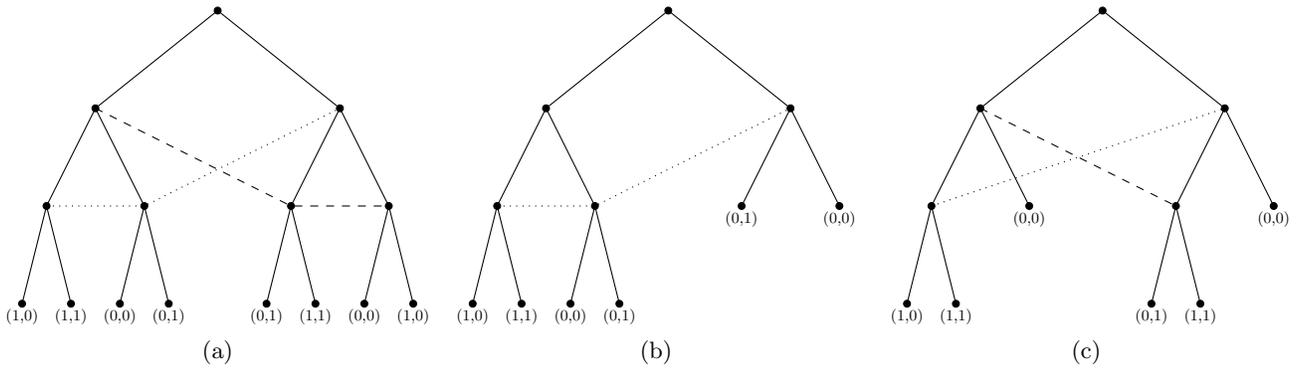
Figure 1: Three examples. The roots are Chance nodes where Chance chooses its move uniformly at random. Dashed information sets belong to player 1 and dotted ones to player 2. The node in game (b) that forms its own information set belongs to player 1. (b) has an exact deterministic timing, but (a) and (c) do not.

node in the information set. But this cannot be achieved in the game in Figure 1(a), because the left-hand side of the tree ensures that the expectation of the time distribution for player 1's information set must be at least 1 lower than that for player 2's information set, but the right-hand side implies the opposite. Still, we may achieve *something* with randomization. For example, we may draw an integer i uniformly at random from $[N-1] = \{1, \ldots, N-1\}$, offer the first player to move a bet at time i and the second player a bet at time $i+1$. Then, if a player is offered a bet at time 1 or time N, the player will know exactly at which node in the extensive form she is. On the other hand, if she is offered a bet at any time $t \in \{2, \ldots, N-1\}$, she obtains no additional information at all, because the conditional probability of t being the selected time is the same whether she is the first or the second player to move. Hence, as long as $i \in \{2, \ldots, N-2\}$, which happens with probability $(N-3)/(N-1)$, neither player learns anything from the timing. We say the game is *approximately timeable*: we can come arbitrarily close to timing the game by increasing N, the number of time periods used. This immediately raises the question of whether *all* games are approximately timeable, and if so how large N needs to be for a particular approximation.

1.2   Our contribution

In the next section we define *exactly timeable games*, give a characterization of these games, and show that there is a linear-time algorithm that decides whether an extensive-form game is exactly timeable. In Section 3 we define ϵ-timeability and argue that this is the correct definition. In Section 4 we give an example of an onion routing game that is not timeable. This shows that, due to timeability issues, onion routing protocols can only approximately obtain a certain desired property. In Section 5 we show that all extensive-form games are ϵ-timeable for any $\epsilon > 0$, but also that these ϵ-timings can easily become too time-consuming for this universe: for any number r, there exists a game Γ_r such that for sufficiently small ϵ, any ϵ-timing of Γ_r will take time at least $2^{2^{\cdots 2^{\frac{1}{\epsilon}}}}$ where the tower has height r. In Section 6 we ask what happens if we have some control over the players' perception of time. We assume that there exists a constant c such that any player will always perceive a time interval of length t as having length between $\frac{t}{c}$ and ct, and

otherwise we have complete control over the players' perception of time. We show that even under these assumptions, the lower bound from Section 5 still holds.

2.   EXACTLY TIMEABLE GAMES

DEFINITION 1. *For an extensive-form game* [3] Γ, *a deterministic timing is a labelling of the nodes in Γ with non-negative real numbers such that the label of any node is at least one higher than the label of its parent. A deterministic timing is* exact *if any two nodes in the same information set have the same label.*

An exact deterministic timing is the same as the time function in the definition of a chronological order by Weibull [23]. Since we will also be discussing games that cannot be timed, we need this more general definition of timings that are not exact.

This definition allows times to be nonnegative real numbers rather than integers, which makes some of the proofs cleaner. However, given a deterministic timing with real values, one can always turn it into a timing with integer values by taking the floor function of each of the times.

The following theorem says that it is easy to check whether a game has an exact deterministic timing, providing multiple equivalent criteria. Criterion 2 is presumably most useful for a human being looking at small extensive-form games, while criterion 3 is easy for a computer to check.

THEOREM 1. *For an extensive-form game Γ, the following are equivalent:*

1. *Γ has an exact deterministic timing.*

2. *The game tree Γ can be drawn in such a way that a node always has a lower y-coordinate than its parent, and two nodes belong to the same information set if and only if they have the same y-coordinate.*

3. *Contracting each information set in the directed graph Γ to a single node results in a graph without oriented cycles.*

---

[3]For an introduction to the game-theoretical concepts used in this paper, see, for example, [15]
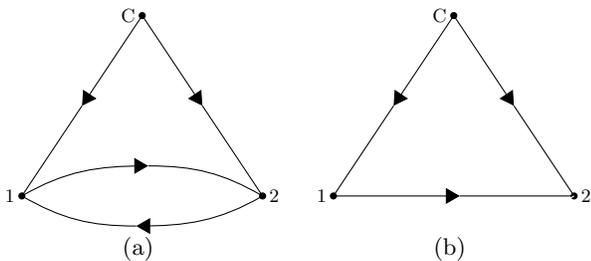
Figure 2: Example of how to use Theorem 1 to test if the extensive-form games (a) and (b) in Figure 1 have exact deterministic timings. The top node is the Chance node, the left node corresponds to player 1's information set, and the right node corresponds to player 2's information set.

PROOF. "1 $\Rightarrow$ 2:" Given an exact deterministic timing (WLOG, with integer-valued times), we draw Γ such that each node has y-coordinate equal to the negative of its time. As the timing is exact, nodes in the same information set have the same y-coordinate. To ensure that any two nodes with the same y-coordinate are in the same information set, we perturb each node based on its information set. This can be done deterministically: for example, if there are q information sets in the game, then subtract i/q from the time of each node in the ith information set.

"2 $\Rightarrow$ 3:" Given such a drawing, contracting each information set results in all edges going downwards, so the resulting graph cannot have directed cycles.

"3 $\Rightarrow$ 1:" The nodes of a directed acyclic graph can be numbered such that each edge goes from a smaller to a larger number. This numbering can be used as a deterministic timing. $\square$

We can use criterion 3 of Theorem 1 to test whether the games in Figure 1(a) and 1(b) are timeable. First we draw a node for each information set: One for the root, one for player 1's information set and one for player 2's information set. (If one of the players had more than one information set, that player would have had more than one node in the contracted graph.) We ignore the leaves, as they can never form cycles. In the games in Figure 1(a) and 1(b), we can get from the root to each of the two players' information sets, so we draw a directed edge from the root to each of the two other nodes. We can also get from player 1's information set to player 2's, and in the game in Figure 1(a) we can go from player 2's information set to player 1's. When we draw these directed edges (without multiplicity) we get Figure 2(a) and Figure 2(b), respectively. We see that the graph in Figure 2(a) has a cycle, so the game in Figure 1(a) is not exactly timeable, while graph in Figure 2(b) does not have a cycle, so the game in Figure 1(b) is exactly timeable. The contracted graph can be constructed in linear time, and given this directed graph, we can in linear time test for cycles [3, Section 22.4]. Thus, we can test in linear time whether a game is exactly timeable.

3. ϵ-TIMEABILITY

We now move on to approximate timeability.

DEFINITION 2. The total variation distance (also called statistical distance) between two discrete random variables X_1 and X_2 is given by

$$\delta(X_1, X_2) = \sum_x \max(\Pr(X_1 = x) - \Pr(X_2 = x), 0)$$

where the sum is over all possible values of X_1 and X_2. This measure is symmetric in X_1 and X_2. If $\delta(X_1, X_2) \leq \epsilon$ we say that X_1 and X_2 are ϵ-indistinguishable.

A (randomized) timing is a discrete distribution over deterministic timings. For a game, a timing of the game, a player and a node v belonging to that player, the player's timing information at v denoted $X_{\preccurlyeq v}$ is the sequence of times X_u for nodes u belonging to that player on the path from the root to v (including v itself). Thus, for a fixed game, timing, player, and node, the timing information is a random variable.

The timing is an ϵ-timing if for any two nodes u and v in the same information set, $\delta(X_{\preccurlyeq u}, X_{\preccurlyeq v}) \leq \epsilon$. A 0-timing is also called an exact timing.

A game is exactly timeable if it has an exact timing, ϵ-timeable if it has an ϵ-timing, and approximately timeable if it is ϵ-timeable for all $\epsilon > 0$.

The following proposition implies that Γ being exactly timeable is equivalent to each of the three criteria in Theorem 1.

PROPOSITION 2. A game is exactly timeable if and only if it has an exact deterministic timing.

PROOF. An exact deterministic timing is a special case of an exact randomized timing. Conversely, given an exact randomized timing of a game, we can label each node with its expected time to obtain an exact deterministic timing. $\square$

We will show that in fact all games are approximately timeable.

3.1 Justification of definition

As stated in [18] and [7] the total variation distance $\delta(X_1, X_2)$ can be seen as a betting advantage: Suppose you are given a random value X_I were I is uniformly distributed on $\{1, 2\}$ independently from X_1 and X_2. You then bet on whether $I = 1$ or $I = 2$. If you guess correctly you get utility 1 and otherwise you get -1. If you play this game optimally, your expected utility is $\delta(X_1, X_2)$. More generally, if a player is playing a game Γ and is in an information set with two nodes 1 and 2 which have different optimal actions, we can think of this as the player betting on which node she is in. For this reason, total variation distance is a good measure of how much a game is distorted by side information. This is captured by the following theorem.

THEOREM 3. Let Γ be an extensive-form game with perfect recall and utilities in $[0, 1]$ where player i has at most m nodes in any history, and let X be an ϵ-timing. If σ_i' is a player i strategy that uses the timing information,[4] there is a strategy σ_i that does not use timing information, such that for any strategy for the other players σ_{-i}', which may also use the timing information, we have $u_i(\sigma_i', \sigma_{-i}') - u_i(\sigma_i, \sigma_{-i}') \leq m\epsilon$.

---

[4] Formally, this is a strategy in the extensive-form game Γ' where the first move is a chance move which gives all the randomness in X, and all the next moves are as in Γ. Two nodes belong to the same information set in Γ' if they belong to the same information set in Γ and have the same timing history.

The proof of this theorem, as well as later theorems, can be found in the full version [8]. Conversely, an example, also in the full version, shows that the $m\epsilon$ cannot, for any m and ϵ, be improved to something better than $1 - (1 - \epsilon)^m$, which for small ϵ is approximately $m\epsilon$. By using the above theorem on each player in a game we get this corollary.

COROLLARY 4. *Let Γ be a perfect recall game with utilities in $[0, 1]$ where each player has at most m nodes in any history. If Γ' is the game Γ with timing information X, where X is an ϵ-timing, then any Nash-equilibrium σ of Γ is an $m\epsilon$-approximate Nash equilibrium of Γ'.*

One possible criticism of the definition of ϵ-timings is that it is only about the advantage the players get *on average*. Another definition would be to require that with probability $1 - \epsilon$ the players learn nothing at all from the timing information. We say a timing X is ϵ-*ex-post-perfect* if for any node u there is probability at least $1 - \epsilon$ that the timing information $X_{\preccurlyeq u}$ at u takes a value $x_{\preccurlyeq u}$ such that for all v in the same information set as u we have $\Pr(X_{\preccurlyeq u} = x_{\preccurlyeq u}) = \Pr(X_{\preccurlyeq v} = x_{\preccurlyeq u})$. The following theorem shows that this definition would give essentially the same results as our definition of ϵ-timeablility.

THEOREM 5. *Let Γ be an extensive-form game with perfect recall. If X is a ϵ-ex-post-perfect timing of Γ then X is an ϵ-timing of Γ. Conversely, there exists a constant c_Γ such that for all ϵ if X is an ϵ-timing of Γ using time at most N then there exists a $c_\Gamma\epsilon$-ex-post-perfect timing of Γ that uses time at most $2N + 1$.*

The first part of the theorem is obvious from the two definitions. The intuition in the second part is to modify the ϵ-timing X to get a $c_\Gamma\epsilon$-ex-post-timing X': We can assume that all times in X are integers. With high probability, the times in X' will just be twice the time in X. However, in cases where X gives away some probabilistic information (this could happen in all cases), X' will, with small probability take an odd value instead. This is done in such a way that given that the time is even, we have $\Pr(X'_{\preccurlyeq u} = x_{\preccurlyeq u}) = \Pr(X'_{\preccurlyeq v} = x_{\preccurlyeq u})$.

Another possible criticism of the concept of timeability is that you can always transform a not exactly timeable extensive-form game to its normal form, that is, you ask each player one by one to report what they would do in any possible situation. This normal-form game can be considered to be an extensive-form game where each player only has one move. This game is clearly timeable. However, there are several problems in doing this. First of all you lose the temporal information, so this transformation can change properties of the game which depend on temporal information or on the beliefs the players have during the game. For example, transforming an extensive-form game to a normal-form game will often introduce new sequential equilibria.

A second problem in transforming to normal form, is that conceptually simple modifications of an extensive-form game might correspond to complicated modifications of the normal-form version. One example is from correlated equilibria, where the players get access to some correlated randomness during the game. For each distribution of the randomness, we get an extensive-form game which can be transformed to normal form. Thus, any question about correlated equilibria

of an extensive-form game and be formulated as questions about classes of normal form games, but this would be a different and more complicated question. So although all not exactly timeable extensive-form games can be transformed to normal-form (and hence timeable) games it is possible that some theorems which hold for such modifications of timeable game, do not hold for in general for similar modifications of extensive-form games.

Finally, when transforming an extensive-form game to its normal form, there will generally be an exponential blow-up, both in the amount of communication needed to play the game, and in the description length of the game. This means that unless the game is a small toy example, it will not be feasible to play the game as a normal-form game. This also gives another example of how a theorem can hold for all timeable games, but not necessarily for all extensive-form games: Suppose you care about some function f defined on extensive-form games, and f is *not* affected when you transform a game from its extensive-form version to its normal form. Suppose further that you have a polynomial time algorithm for computing f on timeable games. Clearly, this algorithm can be used to compute f on all extensive-form games as well: You first transform your extensive-form game to a normal-form, and hence timeable, game, and then you apply the function on the transformed game. While this algorithm will correctly compute f, the algorithm will not be computable in polynomial time because of the blow-up in description size.

We have argued that there are many reasons not to transform extensive-form games to normal form, but you could argue that some not exactly timeable games have "equivalent" extensive-form games that can be timed, for some definition of equivalent. For example, in the game in Figure 1(a) one player's choice does not affect the other player's knowledge or options, so you could define an "equivalent" game where first player 1 moves and then player 2 moves. However, we do not see this as a weakness of the definition: When you define the rational numbers, you do not have to give an algorithm for checking if e.g. a fraction between two square roots is rational, even though such an algorithm may be useful. Similarly, in this paper we will not investigate which games that can somehow be "simplified" to timeable games, but we think it will be an interesting question for future research. One of the points in this paper is to argue that when proving theorems about games that have real world applications, you do not lose much by assuming timeability. This point stands: If you prove that a proposition P holds for all timeable games and you have an equivalence relation on all games which behave well with respect to P (meaning that if Γ and Γ' are equivalent then $P(\Gamma) \Leftrightarrow P(\Gamma')$) then P clearly holds for all games that are equivalent to a timeable game.

4. AN ONION ROUTING EXAMPLE

Suppose four players, $\{0, 1, 2, 3\}$ are sending messages to each other. Each player i can only send envelopes to the next player, player $i+1$ (modulo 4), but is only interested in sending a message to the previous player. The player sends these messages by using 3 nested envelopes. For example, if player 2 wants to send a message to player 1 she will write down the message, put it in an envelope marked "1", put that in an envelope marked "0", put that in an envelope marked "3" and hand that to player 3.
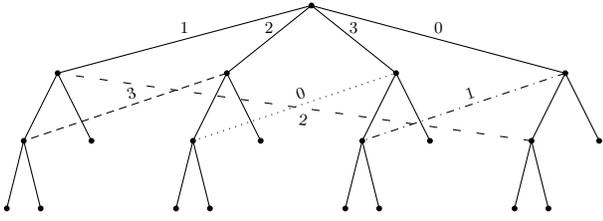
Figure 3: The extensive form of the onion routing game. The root belongs to chance who decides who is going to send a message. Each of the four players has only one information set illustrated by dashed/dotted lines. Using criterion 2 or 3 from Theorem 1 it is clear that this game is not exactly timeable.

We now define a game where first chance chooses one player j. Now player j will nest three envelopes as described above and hand it to $j+1$. Each of the two players on the way, $j+1$ and $j+2$ can choose wether to follow the protocol of opening their envelope and passing on the envelope inside or to obstruct the protocol by keeping the envelope. Player i gets utility -1 if a message went from $i+2$ to $i+1$ and player i gets utility $1+\epsilon$ if a message went from $i+3$ to $i+2$. That is, each player want to obstruct the protocol if they are the second person to pass on an envelope, but wants to help if they are the first.

This defines an extensive-form game with perfect recall as shown in Figure 3, and without any timing information each player will collaborate in the protocol. However, using Theorem 1 it is easy to see that the game is not exactly timeable. One possible way to ϵ-time it is to let X_1 be uniformly distributed on $\{1, \ldots, N-1\}$ and let the sender send the first envelope at time X_1 and and let each of the two helpers make their move one time unit after receiving the envelope. This way the players will follow the protocol, unless they receive an envelope at time N. Thus, the protocol will fail with probability $\frac{1}{N-1}$.

In this example the players used envelopes because they could not send the message directly. However, in the Tor network [4] a similar protocol is instead used to provide anonymity. In this protocol the concern is not so much reliability, but how much the Tor nodes learn about your communication. Because the onion routing game is not exactly timeable, you cannot avoid giving the routers some information about how far they are from the sender of the message. In particular, one router might learn that it is communicating directly with the original sender. As long as there are only messages going one way, the routers might not get much information if the network has been running for years, because this means that N is large.[5] However, if the recipient wants to answer the message within a few seconds, the routers will get more information: Now each router will be passing on two messages within a few seconds and the time gap between the messages will always be smallest for the router closest to the recipient of the first message.

---

[5]If the time a user decides to send a message over the network in not uniformly distributed over the day or the week, the routers would get some information about their position in the path. However, if it takes less than a couple of seconds to send a message through the path, that will only be a very small amount of information.

Notice that this is *not* a problem for the Tor network, as it was never designed to hide this information from the router. However, if we wanted to design an onion routing protocol where the routers do not know how far they are from the end-points, this would be a problem. This is why the concept of timeability is important: you cannot just define an extensive-form game with perfect recall and expect that it can be implemented (even in principle), unless you check that it is timeable.

Except for onion routing related games, we have not been able to find games that are not exactly timeable but would be useful to play in real life. However this only strengthens our thesis that when you want to prove a theorem about all extensive-form games, you do not lose much from assuming timeability.

5. UPPER AND LOWER BOUNDS

From [7] we have the following definition and theorem.

DEFINITION 3. *Let $X_1, \ldots, X_n$ be random variables over $\mathbb{N}$ with some joint distribution such that we always have $X_1 < X_2 < \cdots < X_n$. We say that $(X_1, \ldots, X_n)$ has ϵ-indistinguishable m-subsets if for any two sets in indices $\{i_1, \ldots, i_m\}$, $\{j_1, \ldots j_m\} \subset [n]$ of size m, the two random sets $\{X_{i_1}, \ldots, X_{i_n}\}$ and $\{X_{j_1}, \ldots, X_{j_n}\}$ are ϵ-indistinguishable. We slightly abuse notation and say that $(X_1, \ldots, X_n)$ has ϵ-indistinguishable subsets if for all $m < n$ it has ϵ-indistinguishable m-subsets.*

In the following $\exp_2$ denotes the function $\exp_2(x) = 2^x$, and $\exp_2^n(x)$ denotes iteration of $\exp_2$, so $\exp_2^n(x) = 2^{2^{\cdots^{2^x}}}$ where the tower contains n 2's.

THEOREM 6. *For any $n \in \mathbb{N}$ and any $\epsilon > 0$ there exists a distribution of $(X_1, \ldots, X_n)$ such that $1 \le X_1 < X_2 < \cdots < X_n$ are all integers and X has ϵ-indistinguishable subsets. For fixed n we can ensure that X_n never take values larger than $\exp_2^{n-2}\left(O\left(\frac{1}{\epsilon}\right)\right)$. Conversely, for any such distribution, X_n must take values of at least $\exp_2^{n-2}\left(\Omega\left(\frac{1}{\epsilon}\right)\right)$ for sufficiently small ϵ. This lower bound holds even if we only require the $n-1$-subsets of $(X_1, \ldots, X_n)$ to be ϵ-indistinguishable.*

The following gives intuition for the upper bound on N. For $n = 2$ it is easy to construct (X_1, X_2) that has ϵ-indistinguishable subsets. For example, we can take X_1 to be uniformly distributed on $[N-k]$ for some constants N and k and set $X_2 = X_1 + k$. We can then use a recursive construction for higher n. If $(X_1, \ldots, X_n)$ has ϵ-indistinguishable subsets and consecutive X_i's are usually not too close to each other, we can construct $(Y_1, Y_2 \ldots, Y_{n+1})$ that has ϵ'-indistinguishable subsets for some ϵ'. To do this, we choose Y_1 uniformly between 1 and a sufficiently large number, and choose each gap $Y_{i+1} - Y_i$ uniformly and independently from $\{1, 2, \ldots, 2^{X_i}\}$. For a proof that this works, see [7].

The intuition about the lower bound on N is that for $n = 2$ an $n-1$-subset contains 1 number, and the *size* of this number gives away some information about whether it is the higher or lowest. For $n = 3$ an $n-1$ subset contains two numbers and their *distance* gives away some information about whether it is the middle number or another number that is missing from the set. For $n = 4$ an $n-1$ subset contains 3 numbers, and now the *ratios between the*

two distances gives away information about which number is missing, and so on.

We can use the construction to approximately time any game.

THEOREM 7. *All games with at most m nodes in each history can be ϵ-timed in time $\exp_2^{m-3}\left(O\left(\frac{1}{\epsilon}\right)\right)$. In particular, all games are approximately timeable.*

PROOF. Take any game and $\epsilon > 0$. We want to show that the game is ϵ-timeable. First we find some distribution of $(X_1, \ldots, X_{m-1})$ that has ϵ-indistinguishable subsets. Now we let the time of the root be 0 and the time of a node at depth d be given by X_d. As the X_d's take values in $\mathbb{N}$ and are increasing this gives a timing of the game. If two nodes v and w belong to the same information set, the player i who owns these nodes will have the same number $j-1$ of previous nodes at v and at w. As $(X_1, \ldots, X_{m-1})$ has ϵ-indistinguishable subsets, it has ϵ-indistinguishable j-subsets, so if the root does not belong to player i there is total variation distance at most ϵ between the two nodes' timing information. Similarly, $(X_1, \ldots, X_{m-1})$ has ϵ-indistinguishable $j-1$-subsets, so if the root belongs to player i the total variation distance between the two nodes' timing information is also at most ϵ. □

Unfortunately the above upper bound on the time needed to ϵ-time is beyond astronomical even for moderate values of m and ϵ. The following lower bounds shows that for any r there are games that take time $2^{2^{\cdots^{\frac{1}{\epsilon}}}}$ to ϵ-time, where the hight of the tower is r.

THEOREM 8. *Given $r \geq 1$ there exists $\epsilon_r > 0$ and a game with $16r + 3$ players and at most $\max(3r, 2r+3)$ nodes per player per history such that for any ϵ-timing of the game with $\epsilon \leq \epsilon_r$ we need time at least $\exp_2^r\left(\epsilon^{-1}\right)$.*

The proof has some similarities to the lower bound part of Theorem 6, but we also need some new ideas. If extensive-form games could have nodes that belong to more than one player the theorem would follow from Theorem 6: We would define a "game" Γ where the first node was a chance node which assigned the numbers from 1 to n to the n players, and the players do *not* learn this number. Then, for each such choice, we would have a node belonging to all but the player who was assigned the number 1, followed by a node belonging to all but the player who was assigned the number 2 and so on. Let X_i denote the time of the i'th node, and suppose that this time is independent from the assignment of numbers to the players. Then a player who was assigned number i gets timing information $(X_1, \ldots X_{i-1}, X_{i+1}, \ldots, X_n)$. If we have an ϵ-timing of this "game", the players should not be able to figure out their assigned number, so the $n-1$-subsets of $(X_1, \ldots, X_n)$ should be ϵ-indistinguishable. Thus, by Theorem 6 this "game" would take time $\exp_2^{n-2}\left(\Omega\left(\frac{1}{\epsilon}\right)\right)$ to ϵ-time for sufficiently small ϵ.

However, in extensive-form games each node only belongs to one player. To fix this problem we could split the node belonging to many players in our "game" to many nodes each belonging to one player to get a real game. For example, if there where three players, we would first have a node belonging to player 2 then one belonging to player 3 (omitting player 1), then one belonging to player 1 and one to player

3 (omitting player 2), and finally one belonging to player 1 and one to player 2. We will denote this game by just listing the order of the nodes: 231312. This strategy does not work, as this game can be ϵ-timed in time $O(\epsilon^{-1})$ by having the three first nodes immediately after each other, then a gap of variable length and then the three last nodes immediately after each other. In this case, with just three players, this problem can be fixed by reordering the nodes to get 233112 which takes time $2^{\Omega(\epsilon^{-1})}$ to ϵ-time. A similar strategy works for four players but this strategy does not seem to work for more than four players.

Instead, the idea is to prove the claim by induction. You take a game Γ that takes time $\exp_2^r\left(O(\epsilon^{-1})\right)$ to ϵ-time and combine it with copies of the four-player game with node order given by 243314412213, where the four-player games and Γ are played by different players. The nodes from Γ and the nodes from the four-player game are ordered in a way that ensures that if the four-player games are ϵ-timed, then the gaps between certain points in Γ will either all get larger and larger or all get smaller and smaller (in both cases by at least some factor greater than 1). This means that if you take the logarithms of certain gaps between node times you get a timing of Γ. As Γ takes time $\exp_2^r\left(O(\epsilon^{-1})\right)$ to ϵ-time, this means that the new game must take $\exp_2^{r+1}\left(O(\epsilon^{-1})\right)$. This is the idea in the proof of Theorem 8, but the actual proof is more complicated because you need a stronger induction hypothesis.

Theorem 8 as stated only shows that the players can get some ϵ advantage if they have unbounded computational power. So we might hope that we could find a timing such that a player with bounded computational power cannot get an ϵ advantage. However, we will now see that this is not possible. The proof of Theorem 8 shows that for any r there exists a game Γ_r, such that for any timing X of the game using time at most N, there is a player i who will be able to earn some ϵ amount in expectation by betting on a *fair bet*. Here a *fair bet* means a bet about which branch of the game that was played, such that the player would have expectation 0 if he did not have the information from the timing, and where the player can win or lose at most 1.

Given Γ_r and N we can define a new two-player game Γ_T between a player master-mind and a scheduler. Here the player master-mind is just a useful abstraction of all the players. To play Γ_T, the player master-mind secretly chooses one player i from Γ_r who is going to make a fair bet, and a strategy which given the available timing information chooses a fair bet. At the same time the scheduler chooses the timing X of Γ_r. Now the game Γ_r is played and player i gets some timing information from X. Then player i makes a fair bet according to the strategy chosen by the player master-mind. The player master-mind gets the utility which the player wins from the bet, and the scheduler gets minus this utility.

Now Γ_T is a two player zero-sum game, and for a fixed value of N there is only a finite number of pure strategies for each player, so we can use the Minimax Theorem. Now Theorem 8 shows that if the scheduler commits to some randomised strategy, there is a move that gives the player master-mind a utility of ϵ. By the Minimax Theorem this implies that there is a randomised strategy for the player master-mind which guarantees a utility of at least ϵ. As this strategy is fixed, the players only need a certain amount of

computational power to follow it, no matter how complicated a strategy the scheduler chooses.

6. IMPERFECT TIMEKEEPING

Previously we assumed that at any time all the players knew the exact time. In practice, this is not a realistic assumption. Even our model of time—that there exists an absolute time, and that everybody's time goes at the same speed—has been proven wrong by relativity theory. If the players cannot feel acceleration, one could use the twin paradox to time games that otherwise cannot be exactly timed [6].[6] A more down-to-earth objection is that it might be possible to affect humans' or even computers' perception of time if you control their environment. The purpose of this section it to show that our lower bounds are quite robust: even if we can determine the players' perception of time within some reasonable bounds, there are games that take a long time to ϵ-time. We will assume each node occurs at some "official" time, x, and that we can also decide the players' perception y of that time. The following definition models a situation where a time interval of length t can be perceived as anything between $l(t)$ and $u(t)$ and where the players do not know when the game started.

DEFINITION 4. *Let $l, u : \mathbb{R}^+ \to \mathbb{R}^+$ be weakly increasing functions satisfying $l(t) \leq t \leq u(t)$. A deterministic $[l, u]$-timing of a game Γ is an assignment of a tuple (x_v, y_v) (two nonnegative real numbers) to each node v such that:*

1. *If we label Γ with just the x_v values we have a deterministic timing of Γ.*

2. *If v and w are two nodes belonging to the same player and v is on the path from the root to w then $l(x_w - x_v) \leq y_w - y_v \leq u(x_w - x_v)$.*

An $[l, u]$-timing is a distribution over deterministic $[l, u]$-timings. The timing information *of player i at a node w given an $[l, u]$-timing consists of the* perceived times, y_v, *of all nodes v belonging to that player between the root and w. Now an $(\epsilon, [l, u])$-timing is an $[l, u]$-timing such that for any two nodes belonging to the same information set, the current player's timing information at the two nodes has total variation distance at most ϵ. An $[l, u]$-timing is an* exact $[l, u]$-timing *if it is a $(0, [l, u])$-timing.*

We now show that even if we can affect the players' clocks by some large constant factor c, there still exist games that cannot be ϵ-timed in time $\exp_2^r(\frac{1}{\epsilon})$.

THEOREM 9. *Let c be an integer and let l, u be functions as in Definition 4 and such that $l(x) \geq \frac{x}{c}$ and $u(x) \leq cx$. Then for any r there exists a game $\Gamma_{c,r}$ with $16(2c^4 + r) + 11$ players such that for sufficiently small ϵ any $(\epsilon, [l(x), u(x)])$-timing of $\Gamma_{c,r}$ has to use time at least $\exp_2^r\left(\frac{1}{\epsilon}\right)$.*

PROOF SKETCH. This proof follows the same idea as the induction step of the proof of Theorem 8. We start with a game Γ_r that does not have an ϵ-timing in time less than $\exp_2^r\left(\frac{1}{\epsilon}\right)$. Instead of the four-player game used in the proof of Theorem 8 we now use a $2(4c^4 + 1)$-player game. □

---

[6]The question whether it is possible to implement a not exactly timeable game on players who are equipped with a perfect accelerometer is beyond the scope of this paper.

The next theorem shows that the above is the strongest theorem we can hope for: if we can make the players' clocks go faster or slower by more that a constant factor, we can implement all games.

THEOREM 10. *Let Γ be a game and l, u functions as in Definition 4 with $\frac{u(t)}{l(t)} \to \infty$ as $t \to \infty$. Then Γ is exactly $[l, u]$-timeable.*

7. CONCLUSION

Not every extensive-form game can be naturally implemented in the world. Games with imperfect recall constitute a well known example of this. In this paper, we have drawn attention to another feature that is likely to prevent the direct implementation of the game in the world: games that are not exactly timeable. We gave necessary and sufficient conditions for a game to be exactly timeable and showed that they are easy to check. Most of the technical contribution concerned approximately timing games; we showed that this can always be done, but can require large amounts of time.

Future research can take a number of directions. Does restricting attention to exactly timeable games allow one to prove new results about these games, or develop new algorithms for solving them—as is the case for perfect recall? It is conceivable that the possibility of games that are not exactly timeable has been an unappreciated and unnecessary roadblock to the development of certain theoretical or algorithmic results. Can our techniques be applied to the design of protocols that should not leak information to participants by means of the time at which they receive messages? Are there natural families of games for which we can obtain desirable bounds for the amount of time required to approximately time them?

We have argued that not exactly timeable games cannot be played, but you can transform such a game to a normal-form game. This normal-form game is identical to the original game in some regards, for example it has the same Nash equilibria, but different in others, for example it might have new sequential equilibria. Furthermore, the normal-form game is typically not playable in practice because it takes exponentially more communication. Is it possible to transform a general not exactly timeable game to an "equivalent" timeable game which only requires polynomially more communication that the original? Here, one possible definition of "equivalent" is that the two games give the same game when transformed to normal form, but other definitions might also be relevant. Answers to this question for different definitions of "equivalent" would tell us in what situations it could be useful to assume timeability for certain algorithmic purposes.

8. ACKNOWLEDGMENTS

References

[1] AUMANN, R. J., HART, S., AND PERRY, M. 1997a. The absent-minded driver. *Games and Economic Behavior 20,* 1, 102–116.

[2] AUMANN, R. J., HART, S., AND PERRY, M. 1997b. The forgetful passenger. *Games and Economic Behavior 20,* 1, 117–120.

[3] CORMEN, T. H., STEIN, C., RIVEST, R. L., AND LEISERSON, C. E. 2001. *Introduction to Algorithms* 2nd Ed. McGraw-Hill Higher Education.

[4] DINGLEDINE, R., MATHEWSON, N., AND SYVERSON, P. 2004. Tor: The second-generation onion router. In *Proceedings of the 13th USENIX Security Symposium.*

[5] HANSEN, K. A., MILTERSEN, P. B., AND SØRENSEN, T. B. 2007. Finding equilibria in games of no chance. In *Computing and Combinatorics.* Springer Berlin Heidelberg, 274–284.

[6] HARTLE, J. B. 2002. *Gravity: An Introduction to Einstein's General Relativity.* Addison Wesley.

[7] JAKOBSEN, S. K. 2015. A numbers-on-foreheads game. arxiv:1502.02849.

[8] JAKOBSEN, S. K., SØRENSEN, T. B., AND CONITZER, V. 2015. Timeability of extensive-form games. arxiv:1502.03430.

[9] KREPS, D. M. AND RAMEY, G. 1987. Structural consistency, consistency, and sequential rationality. *Econometrica: Journal of the Econometric Society*, 1331–1348.

[10] KREPS, D. M. AND WILSON, R. 1982. Sequential equilibria. *Econometrica: Journal of the Econometric Society*, 863–894.

[11] KROER, C. AND SANDHOLM, T. 2014. Extensive-form game abstraction with bounds. In *Proceedings of the fifteenth ACM conference on Economics and computation.* ACM, 621–638.

[12] KUHN, H. W. 1953. Extensive games and the problem of information. *Contributions to the Theory of Games 2,* 28, 193–216.

[13] MILTERSEN, P. B. AND SØRENSEN, T. B. 2008. Fast algorithms for finding proper strategies in game trees. In *Proceedings of the nineteenth annual ACM-SIAM symposium on Discrete algorithms.* Society for Industrial and Applied Mathematics, 874–883.

[14] MILTERSEN, P. B. AND SØRENSEN, T. B. 2010. Computing a quasi-perfect equilibrium of a two-player game. *Economic Theory 42,* 1, 175–192.

[15] NISAN, N., ROUGHGARDEN, T., TARDOS, E., AND VAZIRANI, V. V. 2007. *Algorithmic Game Theory.* Cambridge University Press, New York, NY, USA.

[16] PICCIONE, M. AND RUBINSTEIN, A. 1997. On the interpretation of decision problems with imperfect recall. *Games and Economic Behavior 20,* 1, 3–24.

[17] ROMANOVSKY, J. 1962. Reduction of a game with perfect recall to a constrained matrix game. *Doklady Akademii Nauk SSSR 144,* 62–64.

[18] SPEYER, D. 2008. What is total variation distance? https://sbseminar.wordpress.com/2008/01/14/what-is-total-variation-distance/.

[19] VON STENGEL, B. 1996. Efficient computation of behavior strategies. *Games and Economic Behavior 14,* 2, 220–246.

[20] VON STENGEL, B. AND FORGES, F. 2008. Extensive-form correlated equilibrium: Definition and computational complexity. *Mathematics of Operations Research 33,* 4, 1002–1022.

[21] WAUGH, K., ZINKEVICH, M., JOHANSON, M., KAN, M., SCHNIZLEIN, D., AND BOWLING, M. H. 2009. A practical use of imperfect recall. In *SARA.*

[22] WEIBULL, J. W. 1992. On self-enforcement in extensive-form games. *Games and Economic Behavior 4,* 3, 450–462.

[23] WEIBULL, J. W. 2009. Lecture notes in game theory and economic analysis. Department of Economics, Ecole Polytechnique (Paris), ECO574.