Playing Disjunctive Sums is Polynomial Space Complete

By F.L. Morris, Syracuse

Abstract: A position in a disjunctive sum of games is simply a collection of positions, one from each game: to move in a sum is to move in any one of its constituents. Sums have been studied extensively by Conway and others, and play an important role in Go.

It is shown that the problem of best play in a sum of trivial games is polynomial space complete. Hence it may be conjectured that there is no feasible algorithm for deriving a strategy of play in a sum from knowledge about its constituent games.

1. Introduction

Games of the type of Go, chess, and checkers, when generalized to unbounded sizes, commonly exhibit the property known as polynomial space completeness: that of being as difficult (to play optimally, or to determine the winner from an arbitrary position) as any problems solvable using an amount of scratch storage polynomial in the size of the problem description. [See, for example, Lichtenstein/Sipser; Even/Tarjan; Fraenkel et al.] It is evident that this property of (presumed) intractability depends crucially on the fact that a position of a typical game provides an extremely compact description of the tree of continuations of the game which it allows. If we were to require a game to be represented by its tree written out in full, or even by an acyclic directed graph — representing each reachable position only once, however many lines of play led to it — then clearly an exhaustive analysis to determine the winner could be made to run in time polynomial in this ridiculously inflated input size.

An operation on games which has been extensively studied is the disjunctive sum, introduced by Conway [1976]: a position in the sum of several games is simply a collection of positions, one in each game; to move in a sum one moves in any one of its constituents. Thus Nim, for example, is the sum of several uninteresting games, each played with a single heap of matchsticks. Many games in which the moves may be regarded as the removal of nodes or edges from a graph (e.g., “dots and boxes”) are prone to produce sums as positions from the cutting up of the graph into mutually isolated components. In particular, sums arise profusely in the endgame of Go, and have therefore been studied in practice as well; it is said that professional Go players are able to play the endgame without error.
One is led to wonder whether games whose complication relative to the size of the position arises solely from the formation of sums cannot perhaps be played by some reasonably efficient algorithm. This paper will give strong evidence that this is a vain hope, by showing that the problem of best play in a sum of individually trivial games is polynomial space complete.

Section 2 will define a language for expressing positions in sums of games which individually are written out as trees terminating in numbers. Section 3 will show that the problem of best play in such sums is \( NP \)-hard; this weak result can be shown by a direct transformation from a known \( NP \)-complete problem, and will bring out most of the ideas in game construction which we will need. Section 4 will, by a slight elaboration of the construction, and a more lengthy transformation from a known \( P \)-space complete problem, show that in fact we have \( P \)-space completeness. Section 5 will discuss the applicability of the present result to Conway's purely combinatorial notion of games in which numbers are not taken as primitive.

2. Definition of the Problem

A noteworthy feature of games which are sums is that in any single component one player may make several consecutive moves (the intervening moves by his opponent being made in other components). Motivated presumably by a desire to regard the formation of sums as an operation building up games from other games, Conway has been led to banish the notion of necessary alternation of moves from his definition of games. That is, rather than regard a game in the abstract as a tree with two shapes of nodes which alternate along any path from the root (and show which player's turn it is), he in effect provides two kinds of edges: for him the general form of a game (or position, which is now the same thing) \( G \) is

\[
\{G_1^L, \ldots, G_m^L | G_1^R, \ldots, G_n^R \}.
\]

Here the \( G_i^L \), called the left options of \( G \), are other games (positions) to which one player (correspondingly called Left) may transform \( G \) by a single move, if it is Left who in fact moves in \( G \). Symmetrically, the \( G_i^R \), or right options, denote the possible moves by the other player, called Right, in \( G \).

In this paper we shall be considering games described as sums of trees of this sort. To actually play such a game, one must additionally specify which player starts; play in the game as a whole then does proceed strictly by alternating moves.

Conway is able to erect his entire universe of games on nothing more than the inductive definition:

If \( L, R \) are any two sets of games, then there is a game \( \{L | R\} \).

Any game is built, ultimately, from empty option sets, and it is uniformly decreed that a player whose turn it is to move, but who has no options to choose from, loses. Here it will be more convenient to allow also numbers as primitive (i.e. terminated) games. These may be thought of as amounts of money to be paid to Left; thus Left will prefer