

sci.math: Re: How long would it take a computer to completely "solve" chess?

Re: How long would it take a computer to completely "solve" chess?

Source: <http://sci.tech-archive.net/Archive/sci.math/2004-09/3661.html>

From: Guy Macon (<http://www.guymacon.com>)

Date: 09/15/04

Date: Wed, 15 Sep 2004 15:18:16 -0700

(I am on a hot project and no longer have time to respond to or even read all the posts in this newsgroup)

Sean O'Leathlobhair <jwlawler@yahoo.com> says...

>

>Guy Macon <<http://www.guymacon.com>> wrote...

>

>> Sean O'Leathlobhair <jwlawler@yahoo.com> says...

>>

>> >But if we do any pruning, can we be sure that we have not missed some
>> >very obscure trick? We may have solved the game for all practical
>> >purposes but if we have pruned anything, could we really say that we
>> >have theoretically solved chess?

>>

>> Yes. Your objection is valid for the normal pruning that chess
>> programs do, but there are other ways to prune which have zero
>> chance of missing something subtle like throwing away two rooks
>> and a queen in order to set up a forced mate in 49. Here is how:

>>

>> Imagine that you are a program searching the entire move tree,
>> looking to find wins, losses and draws.

>>

>> Now add a modification; if you see a position that has a king and
>> a rook against a king, search one ply farther to rule out stalemate
>> or losing the rook, mark it as a win for the side with the rook,
>> and stop searching that branch. Why? because there exists an
>> algorithm that wins every time from that position. You just pruned
>> without any chance of missing something subtle.

>>

>> Now imagine a thousand years of humans and computers searching
>> for more algorithms that will reduce the size of the search...

>>

>> It's not just that RK vs K position that you can prune at either;
>> you can prune at any position that has 6 men on the board (soon it
>> will be 8); we know the result from every such position already.

Re: How long would it take a computer to completely "solve" chess?

sci.math: Re: How long would it take a computer to completely "solve" chess?

>

>Thanks.

>

>I can see that this sort of pruning is safe. I did not intend to
>question all sorts of pruning, just alpha-beta-pruning (I should have
>been clearer).

One additional thought; the fact that a good chess program finds good moves (moves that lead to positional and material advantages but not to a proven win) would seem to be of little use in solving chess, where you have to search to a proven win, but those good moves have a use; search them first. You might find a winning line faster that way.

Here is what a solution to chess would look like assuming that the answer is a win for white;

One move by white (the best move from the initial position)

All possible replies by black, no matter how stupid.

One move (the best move) by white for each of the black moves.

...and so on.

At many points, the tree will get to the same position (with same castling, en passant, etc. state) via two paths. combine the paths when this happens.

Stop searching when one of the following is found:

Checkmate

A position in the tablebase that is a forced win.

A position that has a known algorithm for winning.

(Some of your computing power should be spend making the tablebase larger and looking for new algorithms for winning certain positions.)

The resulting tree is still way too large for any computer we will have in the next few years, but the resulting tree is far, far smaller than the tree of all possible moves.

--

Guy Macon <<http://www.guymacon.com>>