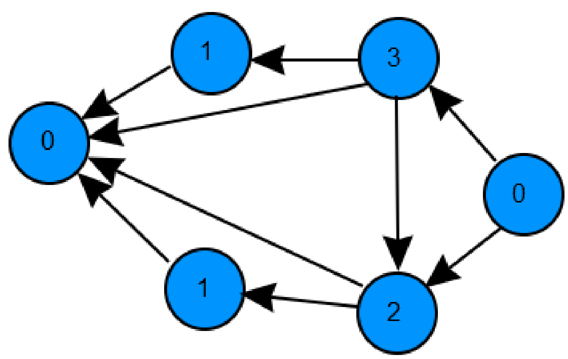
Nim and Impartial Games

April 23, 2017

- Applies to non-infinite, two-player Impartial Games (moves depend only on the position and not on who is moving)
- For simplicity assume loser is the player who can't make a move
- Let's define the nim-value for each node on the game graph as $g(v) = \min\{n \geq 0 : n \neq g(u) \text{ for } u \text{ in } \mathrm{out}(v)\}$. For example:



  Note nodes with nim-value 0 are losing while everything else is winning

- When you have a large game consiting of making a single move in subgames $G_1, \ldots, G_k$, then if the state node v corresponds to the node combination $v_1, \ldots, v_k$, then the nim value is

$$g(v) = g_1(v_1) \oplus \ldots \oplus g_k(v_k)$$

  Here $\oplus$ is bitwise xor. It's correct because:

  1. No next state has the same nim-value. First note that if the move was made in subgame i to v_i', then

$$\begin{aligned} g(v') \oplus g(v) &= [g_1(v_1) \oplus \ldots \oplus g_i(v_i') \oplus \ldots \oplus g_k(v_k)] \\ &\quad \oplus [g_1(v_1) \oplus \ldots \oplus g_i(v_i) \oplus \ldots \oplus g_k(v_k)] \\ &= g_i(v_i') \oplus g_i(v_i) \end{aligned} \tag{1}$$

Suppose by contradiction that there exists such move to v' that $g(v) = g(v')$. In that case $g(v) \oplus g(v') = g_i(v_i') \oplus g_i(v_i) = 0 \Rightarrow g_i(v_i') = g_i(v_i)$ which contradicts the definition of g_i.

2. This $g(v)$ is minimal. This can be proven by induction. For terminal nodes it's obviously true. Now let's look at some node v, and assume that for each connected node v', the nim-value $g(v')$ is $g_1(v_1') \oplus \ldots \oplus g_k(v_k')$. Let's look at some non-negative value $a < g(v)$. If $g(v') = a$, then using (1) for some i we have:

$$g_i(v_i) \oplus g_i(v_i') = a \oplus g(v) \Rightarrow g_i(v_i') = g_i(v_i) \oplus (a \oplus g(v))$$

Since $a < g(v)$, in the leftmost bit j where they differ, we must have $a_{(j)} = 0$ and $g(v)_{(j)} = 1$. Pick i such that $g_i(v_i)_{(j)} = 1$. Since the leftmost 1-bit of $a \oplus g(v)$ is j, we have that:

$$g_i(v_i) \oplus (a \oplus g(v)) < g_i(v_i) \Rightarrow g_i(v_i') < g_i(v_i)$$

Due to the definition of g_i, such v_i' must exist. Since the game tree is a DAG, we can order the game-state nodes and easily apply induction.