You Just Got Yaniv'd - GRP_3

Aiden Sibley (20ajs18@queensu.ca)
Ethan Fighel (20eaf4@queensu.ca)
Oliver Dantzer (oliver.dantzer@queensu.ca)
Will Wu (wu.will@queensu.ca)

Abstract

The Game Yaniv: `https://en.wikipedia.org/wiki/Yaniv_(card_game)`

Summary:
The game is played with a standard deck of 52 cards. Each card represents a digit from 1-13, where an ace would represent 1 and a king would represent 13. Each player of the game is dealt 5 cards to begin. The remaining cards are placed in a pile face down. To begin the game a single card is faced upwards creating a new pile. The goal of the game is to be the first person with cards that sum to 7 or less. To do this players on their turn must replace their cards. Each players hand can be split into one of 2 groups.

1. Cards in a sequence containing the same suit
2. Same card in different suits.

A single card or a group of cards satisfying these requirements can be replaced with either the existing known flipped card already present, or a new card can be taken from the deck. To reiterate, the user has two choices to replace a single card or group of cards (as previously outlined). Note that the player will only substitute a group for one card back, so it is possible to have a hand less than 5.

Model Exploration/Problem Statement: a single scenario of a hand given the players perspective. This includes the 5 or less cards their values and the cards that have been placed upwards. This frame of the game can be represented by a decision tree and logical arguments.

Propositions

e_{ns} = true if a card in hand has value n and suit s
k_i = true if hand has an i number of cards.
p_{ns} = true if piled card is value n and suit s
w = true if the sum of n's for all e_{ns} ¡= 7.
m_{nq} = e_{ns} has q cards with same value n in hand (group type 1 a)
sm_{nq} = p_{ns} has q cards with same value n that are in hand (group type 1 b)
f_{sc} = e_{ns} has c consecutive cards with same s in hand (group type 2 a)
sf_{sc} = p_{ns} has c consecutive cards with same s in hand (group type 2 b)
g_d = true if a group in hand has sum of n equal to d. e.g 2 hearts, 3 hearts, 4 hearts is a group with sum of n equal to 9.
d_g = true if g_d (or just g denoting this specific group) is the highest group in

terms of d in the hand.
sg_d = true if a group in hand+pile has sum of n equal to d.
sd_g = true if sg_d is the highest group in terms of sum d in hand+pile.
c = true when an d_g and a sd_g have at least one e_{ns} in common.
(e.g 2 hearts, 3 hearts, 4 hearts is sd_g and 4 hearts is in hand, and then 4 hearts, 4 diamonds, and 4 spades is d_g)
vd = true when sd_g has higher sum d than d_g.
rx = d_g is put down on top of pile.
ry = best under d_g in hand is put down on top of pile.
us = true if pile card is picked up (option 1)
ud = true if random card from deck is picked up (option 2)
$l = p_{ns}$ less than 7

Constraints

Card constraints for a 52 card deck:

$1 \leq i \leq 5$, so the number of cards in hand is in between 1-5.
$1 \leq n \leq 13$, so n can only be ace, 1, 2,...,10, jack, queen, king.
$1 \leq s \leq 4$, so n can only be diamonds = 1, clubs = 2, hearts = 3, spades = 4.

Every card has a unique suit and value combination assigned to it: e.g there can only be one e_{71}.

Yaniv Game constraints using a 52 card deck:
$1 \leq q \leq 3$, so the number of cards with the same n as a given e_{ns} or p_{ns} is limited to 1, 2 or 3 others.
$2 \leq c \leq 4$, so the number of cards consecutive to a given e_{ns} or p_{ns} with the same s is limited to 2, 3, or 4 others.
$g_d <-> (m_{nq} \wedge f_{sc} \wedge e_{ns} \wedge m_{nq} \wedge f_{sc} \wedge e_{ns} \wedge m_{nq} \wedge f_{sc} \wedge f_{sc} \wedge e_{ns} \wedge m_{nq} \wedge e_{ns})$
(these three things are just groups, categorized because they have differeSnt constraints placed on them, similar to one below)
$sg_d <-> (sm_{nq} \vee sf_{sc} \wedge sm_{nq} \wedge sf_{sc})$
$d_g \rightarrow g_d$ (in order for g_d to be the highest, g_d needs to exist, similar to one below)
$sd_g \rightarrow sg_d$
$w \rightarrow e_{71} \vee e_{72} \vee ... \vee e_{74} \vee (e_{51} \wedge e_{11}) \vee$ ... or any combination of cards in hand that adds up to a sum of $n \leq 7$.
$c \rightarrow sd_g \wedge d_g$
$ry \longleftrightarrow c \wedge vd$
rx $\longleftrightarrow !c$
$l \vee ry \wedge g_d \wedge d_g \wedge rx \wedge sg_d \wedge sd_g) -> us$
$!l \longleftrightarrow ud$

Model Exploration

Originally, to model the constraints.

First-Order Extension

For first-order extension, the universe of discourse, depending on the scenario, could be a standard 52 card deck or the cards in a scenario (hand + face up card). Each variable would be arbitrary card.
Originally, the propositions and constraints relied on subscripts to identify individual elements in the universe of discourse. However, the implementation of predicate logic allows us to use the $\forall$ and $\exists$ quantifiers to effectively do the job of the original subscripts.

Variables subject to change include: card number, card suit, and/or card position in deck.

Example 1: If we wanted to model the existence of a hand with cards 2C, 3H, 5S, 7S, and 9S, we could build a model by doing the following:
Domain(x) = {1-13}
Range(y) = {C,S,D,H}
N = 2,3,5,7,9
s = C, H, S $\forall$

Useful Notation

Feel free to copy/paste the symbols here and remove this section before submitting.

$$\wedge \quad \vee \quad \neg \quad \rightarrow \quad \forall \quad \exists$$

Requested Feedback

What can we do to improve the constraints we have? How can we make better use of logical operations in our draft?