Project Summary

The project models the rules of chess on a standard 8-by-8 board, as played by player Black, who has only a single King and is currently taking their turn. The game ends when Black takes their turn, being neither checkmated nor stalemated.

Propositions

BK1 = can the king move north-west?
BK2 = can the king move north?
BK3 = can the king move north-east?
BK4 = can the king move west?
BK6 = can the king move east?
BK7 = can the king move south-west?
BK8 = can the king move south?
BK9 = can the king move south-east?

BK_No_Moves = True if the king cannot move

Check = True if there is a white piece capable of taking the black king

Checkmate = True if the king is incapable of taking a valid move and is threatened by a white piece.
Stalemate = True if the king is incapable of taking a valid move but is not threatened by a white piece.

Constraints

List of constraint types used in the model and their (English) interpretation. You only need to provide one example for each constraint type: e.g., if you have constraints saying “cars have one colour assigned” in a car configuration setting, then you only need to show the constraints for a single car. Essentially, we want to see the pattern for all of the types of constraints, and not every constraint enumerated.

¬BK1 ∧ ¬BK2 ∧ ¬BK3 ∧ ¬BK4 ∧ ¬BK6 ∧ ¬BK7 ∧ ¬BK8 ∧ ¬BK9 ⇔ BK_No_Moves

BK_No_Moves ⇔ (Checkmate ∨ Stalemate)
If the Black king is unable to move, the game has ended in either checkmate or stalemate.

Checkmate ⇔ ¬Stalemate
Checkmate and stalemate are unable to coexist - the game ends with either one or the other.

Check ∧ BK_No_Moves ⇔ Checkmate

If the Black king is in check when the game-ending scenario occurs (the Black king is unable to move), the game ends in checkmate.

Model Exploration

Plans for the future after the model is functional:

- Experimented with the idea of letting Black play with additional pieces (ie black can have pawns), however the model becomes explosively more complex with the addition of more pieces that can move to block or capture Black's way out of a mate situation.

- Once the model is functional, it may also be possible to add more white pieces to the model in order to simulate a more complicated game.

First-Order Extension

A way to extend the model into predicate logic would be to extend our propositions and constraints to accommodate a board of unlimited size, or to use first-order logical constraints to accommodate arbitrary numbers of any White piece on the board.

Request Feedback

- We are very confused about what jape proofs are needed/how to utilize jape for this project. Right now we have 2 proofs that use our predicates from after the black king has been determined to be unable to more, to prove that given the black king cannot move, and the black king is in check it concludes the black king is in checkmate. With the same predicates, with the exception of "check" being negated, we proved that the king is in stalemate. However in this process, neither proof used "Checkmate ⇔ ¬Stalemate". Does this mean we can remove this as a predicate because it wasn't needed in the jape proofs? What, in general, should we have for jape proofs? (Very confused here)