для тирозина, когда его концентрация в 3,5 раза превышает концентрацию фенилаланина. Полученные результаты доказывают работоспособность предложенной методики определения концентраций веществ в смеси с помощью метода нейронных сетей.

Дальнейшее развитие работы видится в формировании более емкой обучающей выборки, снижении концентрации определяемых аминокислот, расширении числа компонентов в исследуемой смеси и нахождении путей снижения погрешностей их определения с применением метода искусственных нейронных сетей.

Библиографический список

1. *Дворкин, В.И.* Метрология и обеспечение качества количественного химического анализа / В.И. Дворкин. М.: Химия, 2001. С. 261.
2. *Эсбенсен, К.* Анализ многомерных данных / К. Эсбенсен. Черноголовка: ИПХФ РАН, 2005. С. 158.
3. *Хайкин, С.* Нейронные сети / С. Хайкин. М.; СПб; Киев: Вильямс, 2006. С. 219–221, 279.
4. *Тиц, Н.* Энциклопедия химических лабораторных тестов / Н. Тиц. М.: Лабинформ, 1997. С. 960.
5. *Ясницкий, Л.Н.* Введение в искусственный интеллект / Л.Н. Ясницкий. М.: Академия, 2005. С. 37–39.
6. *Аравин, О.И.* Применение нейронных сетей для распознания и классификации патологий в сосудах / О.И. Аравин, А.В. Малыгин // Методы компьютерной диагностики в биологии и медицине: материалы ежегодной науч. школы-семинара 2008 г.; под ред. проф. Д.А. Усанова. Саратов: Изд-во Сарат. ун-та, 2008. С. 51.
7. *Астахова, И.Ф.* Системы искусственного интеллекта / И.Ф. Астахова, А.С. Потапов, В.А. Чулюков. М.: Бином. Лаборатория знаний, 2008. С. 104.

УДК 519.6

О РЕШЕНИИ ШАХМАТНЫХ ПОЗИЦИЙ С ПОМОЩЬЮ ФОРМУЛ ЛОГИКИ ВЕТВЯЩЕГОСЯ ВРЕМЕНИ

Р.В. Хелемендик

Институт прикладной математики им. М.В. Келдыша РАН,
Москва, отдел теоретической математики
E-mail: romash@keldysh.ru

В работе по произвольной шахматной позиции описано построение 4 формул логики ветвящегося времени. Как минимум, одна из этих формул выполнима, и по ее модели строится решение позиции: оценка позиции (ничья или победа одной из сторон), а также необходимая для ее достижения стратегия. Построение по позиции формул и получение модели и решения иллюстрировано примерами.

Ключевые слова: логика ветвящегося времени, выполнимость, модель.

On the Solution of Chess Positions Using Computational Tree Logic

R.V. Khelemendik

Keldysh Institute of Applied Mathematics RAS, Moscow,
Department of Theoretical Mathematics
E-mail: romash@keldysh.ru

The paper describes a construction of four formulas of Computational tree logic corresponding to an arbitrary chess position. At least one of these formulas is satisfiable and leads to the solution of chess position: value of position (a draw or a victory of one of the sides) and necessary strategy for getting this value is constructed using the formula model.

Key words: computational tree logic, satisfiability, model.

ВВЕДЕНИЕ

В настоящей работе представлено построение решения произвольной шахматной позиции. По заданной шахматной позиции строятся 4 формулы логики ветвящегося времени (л.в.в.), соответствующие следующим утверждениям: Θ_1 — белые могут выиграть, Θ_2 — белые могут сделать ничью, но не могут выиграть, Θ_3 — чёрные могут выиграть, Θ_4 — чёрные могут сделать ничью, но не могут выиграть. При этом верны следующие утверждения: либо выполнима формула Θ_1, а остальные формулы невыполнимы; либо выполнимы формулы Θ_2 и Θ_4, а остальные формулы невыполнимы; либо выполнима формула Θ_3, а остальные формулы невыполнимы. В случае выполнимости формулы Θ_i, где $1 \leq i \leq 4$, по модели для неё получается стратегия белых либо чёрных, необходимая для достижения соответствующего результата.

Описываемый в настоящей статье логический подход позволяет теоретически решить любую позицию — установить оценку и получить необходимую для ее достижения стратегию (либо опровергающую стратегию противника). Мы считаем, что одна из сторон побеждает (делает ничью) в том случае, когда она ставит противоположной стороне мат (соответственно добивается пата или троекратного повторения позиции). Вместе с тем при условии нахождения подходящей редуцируемой цели для победы или ничьей (например, выигрыша материала, достижения определённой позиции, занятия или контролирования ключевого поля [1]) такой подход позволяет решить позицию и практически, получив оценку и стратегию (решающую или опровергающую), используя мощности современных компьютеров. Отметим, что считающаяся наилучшей (чемпион мира среди компьютерных программ) в настоящее время компьютерная программа «Рыбка» [2] позволяет просчитывать и анализировать шахматные позиции лишь на некоторую глубину, что, вообще говоря, не всегда позволяет установить оценку позиции и требуемые для её достижения ходы.

Кроме однократного применения описываемого в статье логического подхода для анализа позиций, возможен многократный — метод последовательных приближений, несколько похожий на логический вывод из гипотез. Пусть требуется установить оценку достаточно сложной позиции. Шахматист сначала пробует выяснить, может ли он из этой позиции достичь некоторого класса других позиций, которые ему кажутся более предпочтительными для достижения нужного результата. Затем пробует из них достичь следующего класса более предпочтительных позиций и т.д. Таким образом, глобальная цель (мат или ничья) может быть сведена к последовательности локальных, достижимость которых может быть изучена за приемлемое время. Такой метод является логической реализацией известного в шахматах «мышления схемами» [3], которое с математической точки зрения является разбиением задачи на подзадачи. При этом сложность достижения локальных целей может быть найдена с помощью применения аппарата рекуррентных форм, изложенного в [4].

Л.в.в. в настоящее время изучается в основном в работах зарубежных исследователей и применяется к анализу свойств компьютерных программ [5]. Разрешимость проблемы распознавания выполнимости для этой логики установлена в [6]. Излагаемый в настоящей статье логический подход к решению произвольной шахматной позиции представляет собой на примере шахмат (как модельного объекта) хороший материал для идей и методов применения логики ветвящегося времени. Кроме того, представляет интерес общая структура формул, записывающих позицию, а также вопрос описания классов формул логики ветвящегося времени, которые «эквивалентны шахматам», с тем, чтобы получая шахматное решение, построить по нему модель, которая будет служить решением для другой задачи. Примеры такого подхода содержатся в работе [1], в которой задача сохранения сервера в безопасности системным администратором трансформирована в подходящую шахматную позицию, решение которой даёт искомую стратегию системному администратору.

1. ЛОГИКА ВЕТВЯЩЕГОСЯ ВРЕМЕНИ

В настоящем разделе даются определения формулы логики ветвящегося времени, модели, истинности формулы в вершине модели, выполнимости и общезначимости.

Каждая пропозициональная переменная есть *формула*. Если φ, ψ *формулы*, то θ, являющаяся одним из 12 выражений: $\top$, $(\varphi \wedge \psi)$, $(\varphi \vee \psi)$, $\neg\varphi$, $\forall \bigcirc\varphi$, $\exists \bigcirc\varphi$, $\forall \Box\varphi$, $\exists \Box\varphi$, $\forall \Diamond\varphi$, $\exists \Diamond\varphi$, $\forall (\varphi\mho\psi)$, $\exists (\varphi\mho\psi)$ тоже называется *формулой*. Других *формул* нет.

Модель — это пара $M = \langle \Gamma, L\rangle$, где Γ – это связный ориентированный граф с выделенной вершиной u_0, а L — функция означивания, сопоставляющая каждой вершине множество пропозициональных переменных. *Полным путём* в графе называется бесконечный путь или цепь, последняя вершина которой не имеет сыновей.

Истинность формулы θ в вершине u_i модели M (обозначим это $M, u_i \models \theta$) определяется индуктивно.

— если $\theta = \top$ $[\theta = p]$, то $M, u_i \models \theta$ $[M, u_i \models \theta \rightleftharpoons p \in L(u_i)]$;

— если $\theta = (\varphi \wedge \psi)$ $[\theta = (\varphi \vee \psi)]$, то $M, u_i \models \theta \rightleftharpoons (M, u_i \models \varphi$ и [или] $M, u_i \models \psi)$;

— если $\theta = \neg\varphi$, то $M, u_i \models \theta \rightleftharpoons M, u_i \not\models \varphi$ (неверно $M, u_i \models \varphi$);

— если $\theta = (\varphi \to \psi)$, то $M, u_i \models \theta \rightleftharpoons (M, u_i \models \neg\varphi$ или $M, u_i \models \psi)$;

— если $\theta = \forall \bigcirc\varphi$, то $M, u_i \models \theta \rightleftharpoons$ для каждого сына u_j вершины u_i верно $M, u_j \models \varphi$; либо вершина u_i сыновей не имеет;

— если $\theta = \exists \bigcirc\varphi$, то $M, u_i \models \theta \rightleftharpoons$ существует хотя бы один сын u_j вершины u_i, и верно $M, u_j \models \varphi$;

— если $\theta = \forall\ \Box\varphi$ [$\theta = \exists\ \Box\varphi$], то $M, u_i \models \theta \rightleftharpoons$ для каждого полного пути [существует полный путь] в графе с началом в вершине u_i в каждой его вершине u_j верно $M, u_j \models \varphi$;

— если $\theta = \forall\ \Diamond\varphi$ [$\theta = \exists\ \Diamond\varphi$], то $M, u_i \models \theta \rightleftharpoons$ для каждого полного пути [существует полный путь] в графе с началом в вершине u_i найдется вершина u_j этого пути, для которой верно $M, u_j \models \varphi$;

— если θ есть $\forall\ (\varphi \mho \psi)$ [$\exists\ (\varphi \mho \psi)$], то $M, u_i \models \theta \rightleftharpoons$ для каждого полного пути [существует полный путь] в графе с началом в вершине u_i найдется вершина u_j этого пути, для которой верно $M, u_j \models \psi$, а в каждой вершине u_k этого пути, предшествующей u_j, верно $M, u_k \models \varphi$.

Формула θ *истинна в модели* M, если она истинна в выделенной вершине u_0 этой модели. Формула θ *выполнима*, если она истинна в некоторой модели. Формула θ *общезначима*, если она истинна в каждой модели.

2. ПОСТРОЕНИЕ ФОРМУЛЫ ПО ПОЗИЦИИ. ЗАПИСЬ ИСХОДНОЙ ПОЗИЦИИ

Все 4 формулы Θ_1, Θ_2, Θ_3, Θ_4, которые строятся по заданной шахматной позиции, имеют следующую структуру:

$$Init \wedge Rules \wedge Aim. \tag{1}$$

Формула $Init$ кодирует исходную позицию, т.е. позицию, заданную первоначально и оценку которой вместе со стратегией мы хотим получить. Формула $Rules$ кодирует все возможные ходы, получающиеся из исходной позиции, а также из всех новых получаемых позиций. Формула Aim кодирует цель, например, возможность одной из сторон поставить мат. В данном разделе описывается построение формулы $Init$.

Следуя работе [7], введём 69 переменных: $y_0, y_1, \ldots, y_{68}$, каждая из которых может принимать значения из конечного множества $\{0, 1, \ldots, 12\}$. Будем интерпретировать значения этих переменных следующим образом. Переменная y_0 принимает значения из множества $\{0, 1\}$: в случае $y_0 = 0$ в позиции ход белых, а в случае $y_0 = 1$ в позиции ход чёрных. Переменные $y_1, \ldots, y_{64}$ соответствуют клеткам шахматной доски, занумерованных слева направо и затем снизу вверх, начиная от поля $a1$ (левого нижнего углового поля). Переменная y_i, где $1 \le i \le 64$, принимает значение 0, если i-е поле пусто. При $y_i = 1$ ($y_i = 7$) на поле i находится белый (чёрный) король. При $y_i = 2$ ($y_i = 8$) на поле i находится белый (чёрный) ферзь. При $y_i = 3$ ($y_i = 9$) на поле i находится белая (чёрная) ладья. При $y_i = 4$ ($y_i = 10$) на поле i находится белый (чёрный) слон. При $y_i = 5$ ($y_i = 11$) на поле i находится белый (чёрный) конь. При $y_i = 6$ ($y_i = 12$) на поле i находится белая (чёрная) пешка. Переменная y_{65} (y_{66}) принимает значение 0, если своим последним ходом белые (чёрные) не ходили пешкой на два поля; в противном случае переменная y_{65} (y_{66}) есть число от 1 до 8, соответствующее номеру вертикали, по которой ходила данная пешка. Переменная y_{65} (y_{66}) используется для проверки возможности специального шахматного хода — «взятия пешки на проходе». Переменная y_{67} (y_{68}) принимает значение от 0 до 3, которое соответствует возможности рокировки белых (чёрных) в длинную и/или короткую стороны, либо невозможности рокировки вообще (так, например, если ранее ходил король и участвующая в рокировке ладья, то по шахматным правилам рокировка невозможна).

Шахматную позицию будем называть *легальной* в узком смысле (или просто легальной), если у белых и чёрных по королю и при ходе белых (чёрных) чёрный (белый) король не находится под шахом. Таким образом, всякая позиция, полученная при соблюдении шахматных правил из начальной позиции, с которой начинается шахматная партия, является легальной, а обратное, вообще говоря, неверно. Набор значений переменных $\overline{y} = \langle y_0, y_1, \ldots, y_{68} \rangle$ будем называть *легальным*, если он соответствует легальной позиции.

Для кодирования набора $\overline{y}$ введём 271 пропозициональную переменную $p_0, p_1, \ldots p_{270}$ и установим соответствие между первыми 269 переменными и значениями набора $\overline{y}$ следующим образом. Для удобства записи истинность переменной p_i, где $0 \le i \le 268$, будем обозначать через $p_i = 1$, а ее ложность — через $p_i = 0$. Тогда $p_0 = 0 \Leftrightarrow y_0 = 0$, $p_0 = 1 \Leftrightarrow y_0 = 1$. При $1 \le i \le 66$ значения переменных $p_{4(i-1)+1}, p_{4(i-1)+2}, p_{4(i-1)+3}, p_{4i}$ определяются по значению переменной y_i. Значение $p_{4(i-1)+k}$, где $1 \le k \le 4$ равно значению k-го слева разряда в двоичной записи значения переменной y_i. При $67 \le i \le 68$ значения переменных $p_{265}, \ldots, p_{268}$ определяются по значению переменной y_i:

$p_{264+2(i-67)+k}$, где $1 \le k \le 2$, равно значению k-го слева разряда в двоичной записи значения переменной y_i. Таким образом, набору значений переменных $\overline{y}$ однозначно соответствует набор истинностных значений пропозициональных переменных $p_0, p_1, \ldots p_{268}$. При переходе от значений переменных $\overline{y}$ к пропозициональным переменным коньюнкцию $\bigwedge_{i=0}^{268} p_i^{\alpha_i}$, где $\alpha_i \in \{0,1\}$, а $p_i^{\alpha_i}$ есть p_i при $\alpha_i = 1$, и $p_i^{\alpha_i}$ есть $\neg p_i$ при $\alpha_i = 0$, будем обозначать формулой $\overline{P}(\overline{y})$. Обозначим также через $\overline{y}^0$ набор $\overline{y}$, соответствующий исходной позиции. Тогда

$$Init = \overline{P}(\overline{y}^0).$$

Интерпретация переменных p_{269} и p_{270} будет дана в следующем разделе.

3. ЗАПИСЬ ШАХМАТНЫХ ХОДОВ

В данном разделе строится вторая компонента конъюнкции (1) — формула Rules, записывающая все легальные позиции, получающиеся из исходной позиции, кодируемой формулой Init.

Пусть $p_0 = 0$, т.е. $y_0 = 0$, — в позиции, характеризующейся набором $\overline{y}$, ход белых. Обозначим через $WhiteMoves(\overline{y})$ множество всех легальных позиций, соответствующих позициям после всех допустимых ходов белых, сделанных в позиции $\overline{y}$. Согласно шахматным правилам ход белой фигуры или пешки (в дальнейшем пешки будем также называть фигурами) будем считать допустимым, если он выполнен по шахматным правилам для данной фигуры и после его выполнения белый король не находится под шахом, т.е. поле, на котором он стоит, не будет атаковать ни одна из чёрных фигур. Поскольку всего имеется 6 видов фигур, положим

$$WhiteMoves(\overline{y}) = WhiteKing(\overline{y}) \cup WhiteQueen(\overline{y}) \cup WhiteRook(\overline{y}) \cup$$

$$\cup WhiteBishop(\overline{y}) \cup WhiteKnight(\overline{y}) \cup WhitePawn(\overline{y})$$

и построим эти 6 множеств позиций, соответствующих ходам данного типа фигур.

Однако сначала введём встречающуюся в построении данных множеств формулу $SafeWKing$:

$$SafeWKing(\overline{y}) \rightleftharpoons \exists\, i((1 \le i \le 64) \wedge (y_i = 1) \wedge SafeWKingN(\bar{y}, i) \wedge$$

$$\wedge SafeWKingP(\bar{y}, i) \wedge SafeWKingK(\bar{y}, i) \wedge SafeWKingRQ(\bar{y}, i) \wedge SafeWKingBQ(\bar{y}, i)),$$

которая истинна тогда и только тогда, когда белый король, стоящий на поле i, не находится под ударом ни одной неприятельской фигуры.

Формула $SafeWKingN$:

$$SafeWKingN(\bar{y}, i) \rightleftharpoons \bigwedge_{1 \le l \le 8} SafeWKingNMove(\bar{y}, i, l)$$

истинна тогда и только тогда, когда белый король не находится под ударом черного коня, у которого для этого, вообще говоря, есть не более 8 вариантов направлений для нападения, каждое из которых проверяется формулой

$$SafeWKingNMove(\bar{y}, i, l) \rightleftharpoons \neg(PN(i,l) \wedge (y_{fN(i,l)} = 11)),$$

где $PN(i,l)$ проверяет нападение на белого короля с направлений, близким к крайним вертикалям и горизонталям, а $fN(i,l)$ даёт поле для коня, нападающего на короля с направления l:

$$\begin{array}{ll}
PN(i,1) \rightleftharpoons ((x_v(i) \le 6) \wedge (x_g(i) \le 7)), & fN(i,1) \rightleftharpoons i + 10, \\
PN(i,2) \rightleftharpoons ((x_v(i) \le 7) \wedge (x_g(i) \le 6)), & fN(i,2) \rightleftharpoons i + 17, \\
PN(i,3) \rightleftharpoons ((x_v(i) \ge 2) \wedge (x_g(i) \le 6)), & fN(i,3) \rightleftharpoons i + 15, \\
PN(i,4) \rightleftharpoons ((x_v(i) \ge 3) \wedge (x_g(i) \le 7)), & fN(i,4) \rightleftharpoons i + 6, \\
PN(i,5) \rightleftharpoons ((x_v(i) \ge 3) \wedge (x_g(i) \ge 2)), & fN(i,5) \rightleftharpoons i - 10, \\
PN(i,6) \rightleftharpoons ((x_v(i) \ge 2) \wedge (x_g(i) \ge 3)), & fN(i,6) \rightleftharpoons i - 17, \\
PN(i,7) \rightleftharpoons ((x_v(i) \le 7) \wedge (x_g(i) \ge 3)), & fN(i,7) \rightleftharpoons i - 15, \\
PN(i,8) \rightleftharpoons ((x_v(i) \le 6) \wedge (x_g(i) \ge 2)), & fN(i,8) \rightleftharpoons i - 6.
\end{array}$$

Формула $SaveWKingP$:

$$SafeWKingP(\bar{y}, i) \rightleftharpoons \bigwedge_{1 \le l \le 2} SafeWKingPMove(\bar{y}, i, l),$$

где

$$SafeWKingPMove(\bar{y}, i, 1) \rightleftharpoons \neg((x_g(i) \le 6) \wedge (x_v(i) \ge 2) \wedge (y_{i+7} = 12)),$$

$$SafeWKingPMove(\bar{y}, i, 2) \rightleftharpoons \neg((x_g(i) \le 6) \wedge (x_v(i) \le 7) \wedge (y_{i+9} = 12)),$$

проверяет отсутствие угрозы белому королю со стороны возможной чёрной пешки, нападающей на него с одной из соседних диагоналей. Здесь и далее $x_g(i) = \lfloor (i-1)/8 \rfloor + 1$, $x_v(i) = i - 8 \cdot (x_g(i) - 1)$ — соответственно горизонталь и вертикаль, на которой находится поле с номером i, где $1 \le i \le 64$.

Формула $SafeWKingK$:

$$SafeWKingK(\bar{y}, i) \rightleftharpoons \bigwedge_{1 \le l \le 8} SafeWKingKMove(\bar{y}, i, l)$$

проверяет отсутствие нападения на белого короля со стороны чёрного короля. Необходимость этой формулы вызвана проверкой на легальность позиции после предшествующего хода белого короля. Здесь

$$SafeWKingKMove(\bar{y}, i, l) \rightleftharpoons \neg(PK(i, l) \wedge (y_{fK(i,l)} = 7)),$$

$$\begin{array}{l}
PK(i,1) \rightleftharpoons ((x_v(i) \le 7) \wedge (x_g(i) \le 7)), \quad fK(i,1) \rightleftharpoons i + 9, \\
PK(i,2) \rightleftharpoons ((x_v(i) \ge 2) \wedge (x_g(i) \le 7)), \quad fK(i,2) \rightleftharpoons i + 7, \\
PK(i,3) \rightleftharpoons ((x_v(i) \ge 2) \wedge (x_g(i) \ge 2)), \quad fK(i,3) \rightleftharpoons i - 9, \\
PK(i,4) \rightleftharpoons ((x_v(i) \le 7) \wedge (x_g(i) \ge 2)), \quad fK(i,4) \rightleftharpoons i - 7, \\
PK(i,5) \rightleftharpoons (x_v(i) \le 7), \quad fK(i,5) \rightleftharpoons i + 1, \\
PK(i,6) \rightleftharpoons (x_g(i) \le 7), \quad fK(i,6) \rightleftharpoons i + 8, \\
PK(i,7) \rightleftharpoons (x_v(i) \ge 2), \quad fK(i,7) \rightleftharpoons i - 1, \\
PK(i,8) \rightleftharpoons (x_g(i) \ge 2), \quad fK(i,8) \rightleftharpoons i - 8,
\end{array}$$

и PK проверяет возможность хода короля на крайних вертикалях и горизонталях, а fK даёт новое поле короля после хода, сделанного с поля i в направлении l.

Формула $SaveWKingRQ$:

$$SafeWKingRQ(\bar{y}, i) \rightleftharpoons \bigwedge_{1 \le l \le 4} SafeWKingRQMoves(\bar{y}, i, l)$$

проверяет отсутствие нападения на белого короля чёрной ладьи и ферзя по вертикали или горизонтали. Здесь

$$SafeWKingRQMoves(\bar{y}, i, l) \rightleftharpoons \bigwedge_{1 \le k \le 7} SafeWKingRQMove(\bar{y}, i, l, k),$$

$$SafeWKingRQMove(\bar{y}, i, l, k) \rightleftharpoons \neg(PRI(i, l, k) \wedge PR(i, l, k) \wedge ((y_{fR(i,l,k)} = 8) \vee (y_{fR(i,l,k)} = 9))),$$

где PRI проверяет максимальную длину k вертикали или горизонтали, PR проверяет, что все промежуточные поля этой вертикали или горизонтали пустые, а функция fR даёт поле чёрной ладьи или ферзя с направления l и длины k.

$$\begin{array}{l}
PRI(i,2,k) \rightleftharpoons (x_g(i) - k \ge 1), PRI(i,1,k) \rightleftharpoons (1 \le k \le 8 - x_v(i)), \\
PRI(i,3,k) \rightleftharpoons (x_g(i) + k \le 8), PRI(i,4,k) \rightleftharpoons (1 \le k < x_v(i)), \\
PR(i,2,k) \rightleftharpoons \forall\, h((1 \le h < k) \rightarrow y_{i-8 \cdot h} = 0), \quad fR(i,2,k) \rightleftharpoons i - 8 \cdot k, \\
PR(i,1,k) \rightleftharpoons \forall\, h((1 \le h < k) \rightarrow y_{i+h} = 0), \quad fR(i,1,k) \rightleftharpoons i + k, \\
PR(i,3,k) \rightleftharpoons \forall\, h((1 \le h < k) \rightarrow y_{i+8 \cdot h} = 0), \quad fR(i,3,k) \rightleftharpoons i + 8 \cdot k, \\
PR(i,4,k) \rightleftharpoons \forall\, h((1 \le h < k) \rightarrow y_{i-h} = 0), \quad fR(i,4,k) \rightleftharpoons i - k.
\end{array}$$

Аналогичным образом формула $SaveWKingBQ$

$$SafeWKingBQ(\bar{y}, i) \rightleftharpoons \bigwedge_{1 \le l \le 4} SafeWKingBQMoves(\bar{y}, i, l)$$

проверяет отсутствие нападения на белого короля, слона или ферзя по диагонали.

$$SafeWKingBQMoves(\bar{y}, i, l) \rightleftharpoons \bigwedge_{1 \le k \le 7} SafeWKingBQMove(\bar{y}, i, l, k),$$

$$SafeWKingBQMove(\bar{y}, i, l, k) \rightleftharpoons \neg(PBI(i, l, k) \wedge PB(i, l, k) \wedge ((y_{fB(i,l,k)} = 8) \vee (y_{fB(i,l,k)} = 10))),$$

$PBI(i, 1, k) \rightleftharpoons ((1 \le k \le 8 - x_v(i)) \wedge (x_g(i) + k \le 8)),$
$PBI(i, 2, k) \rightleftharpoons ((1 \le k \le 8 - x_v(i)) \wedge (x_g(i) - k \ge 1)),$
$PBI(i, 3, k) \rightleftharpoons ((1 \le k < x_v(i)) \wedge (x_g(i) + k \le 8)),$
$PBI(i, 4, k) \rightleftharpoons ((1 \le k < x_v(i)) \wedge (x_g(i) - k \ge 1)).$
$PB(i, 1, k) \rightleftharpoons \forall\, h((1 \le h < k) \to y_{i+h+8\cdot h} = 0), \quad fB(i, 1, k) \rightleftharpoons i + k + 8 \cdot k,$
$PB(i, 2, k) \rightleftharpoons \forall\, h((1 \le h < k) \to y_{i+h-8\cdot h} = 0), \quad fB(i, 2, k) \rightleftharpoons i + k - 8 \cdot k,$
$PB(i, 3, k) \rightleftharpoons \forall\, h((1 \le h < k) \to y_{i-h+8\cdot h} = 0), \quad fB(i, 3, k) \rightleftharpoons i - k + 8 \cdot k,$
$PB(i, 4, k) \rightleftharpoons \forall\, h((1 \le h < k) \to y_{i-h-8\cdot h} = 0), \quad fB(i, 4, k) \rightleftharpoons i - k - 8 \cdot k.$

Формула $SaveWKing$ полностью определена. Аналогичным образом строится формула $SafeBKing$, которая истинна тогда и только тогда, когда чёрный король не находится под ударом ни одной белой фигуры.

Построим теперь компоненты множества $WhiteMoves$. Множество позиций, получающихся после хода белого короля — $WhiteKing$, — состоит из позиций, получаемых после обычного хода короля и, быть может, двух позиций, полученных после выполнения рокировок:

$$WhiteKing = \cup_{1 \le i \le 64, y_i = 1}(\cup_{1 \le l \le 8} WhiteKingMove(i, l) \cup WhiteKROOO(i) \cup WhiteKROO(i)).$$

Множество $WhiteKingMove(i, l)$ состоит из не более чем одной позиции $\bar{z}$, соответствующей ходу короля в направлении l:

$$WhiteKingMove(i, l) = \{\bar{z} | PK(i, l), j = fK(i, l), ((8 \le y_j \le 12) \vee$$

$$\vee (y_j = 0)), \forall\, t(((1 \le t \le 68) \wedge (t \notin \{i, j, 65, 68\})) \to z_t = y_t), z_i = 0,$$

$$z_j = 1, z_{65} = 0, z_{68} = fKBO(j, y_{68}), z_0 = 1, SafeKing(\bar{z})\}.$$

При этом новое поле j может быть либо пустым ($y_j = 0$), либо содержать чёрную фигуру, отличную от короля ($8 \le y_j \le 12$). В новой позиции, кроме полей i и j, на остальных полях изменений не происходит. После хода белого короля $z_{65} = 0$, так как данный ход не является ходом белой пешки на два поля. Кроме того, может изменится значение переменной z_{68}, если своим ходом белые лишают чёрных одной из возможностей рокировки, что определяется функцией $fKBO$:

$$fKBO(j, y_{68}) \rightleftharpoons \begin{cases} 2 \cdot \lfloor y_{68}/2 \rfloor, & \text{при } j = 57, \\ y_{68} - 2 \cdot \lfloor y_{68}/2 \rfloor, & \text{при } j = 64, \\ y_{68}, & \text{при } j \notin \{57, 64\}. \end{cases}$$

Рокировка белых в длинную сторону кодируется множеством $WhiteKROOO$, состоящим не более чем из одного элемента:

$$WhiteKROOO(i) = \{\bar{z} | PWKROOO(i), \forall\, t(((1 \le t \le 68) \wedge (t \notin \{1, 3, 4, 5, 65, 67\})) \to$$

$$\to z_t = y_t), z_1 = 0, z_3 = 1, z_4 = 3, z_5 = 0, z_{65} = 0, z_{67} = 0, z_0 = 1, SafeKing(\bar{z})\}.$$

Возможность рокировки белых проверяется формулой

$$PWKROOO(i) \rightleftharpoons ((i = 5) \wedge (y_1 = 3) \wedge (y_{67} - 2 \cdot \lfloor y_{67}/2 \rfloor = 1) \wedge (y_2 = 0) \wedge$$

$$\wedge(y_3 = 0) \wedge (y_4 = 0) \wedge \forall\ t((1 \leq t \leq 68) \to z''_t = y_t) \wedge (z''_0 = 1) \wedge$$

$$SafeKing(\bar{z}'') \wedge \forall\ t(((1 \leq t \leq 68) \wedge (t \notin \{4, 5, 65\})) \to z'_t = y_t) \wedge$$

$$\wedge(z'_4 = 1) \wedge (z'_5 = 0) \wedge (z'_{65} = 0) \wedge (z'_0 = 1) \wedge SafeKing(\bar{z}')),$$

которая истинна лишь в случае, если белый король и левая белая ладья еще не ходили, все три поля между ними свободны, белый король не находится под шахом, не проходит через битое поле и не будет находиться под шахом после рокировки. Рокировка белых в короткую сторону рассматривается аналогично:

$$WhiteKROO(i) = \{\bar{z}|PWKROO(i), \forall\ t(((1 \leq t \leq 68) \wedge (t \notin \{5, 6, 7, 8, 65, 67\})) \to$$

$$\to z_t = y_t), z_5 = 0, z_6 = 3, z_7 = 1, z_8 = 0, z_{65} = 0, z_{67} = 0, z_0 = 1, SafeKing(\bar{z})\},$$

$$PWKROO(i) \rightleftharpoons ((i = 5) \wedge (y_8 = 3) \wedge (\lfloor y_{67}/2 \rfloor = 1) \wedge (y_6 = 0) \wedge$$

$$\wedge(y_7 = 0) \wedge \forall\ t((1 \leq t \leq 68) \to z''_t = y_t) \wedge (z''_0 = 1) \wedge$$

$$\wedge SafeKing(\bar{z}'') \wedge \forall\ t(((1 \leq t \leq 68) \wedge (t \notin \{5, 6, 65\})) \to z'_t = y_t) \wedge$$

$$\wedge(z'_6 = 1) \wedge (z'_5 = 0) \wedge (z'_{65} = 0) \wedge (z'_0 = 1) \wedge SafeKing(\bar{z}')).$$

В позиции, получаемой после рокировки, изменяется содержимое 4-х клеток шахматной доски — обнуляются поля с прежними положениями короля и ладьи, а прежде пустые поля, на которые перемещаются король и ладья, теперь принимают значения 1 и 3. Кроме того, белые уже навсегда лишаются возможности (повторной) рокировки: начиная с этого хода — $z_{67} = 0$.

Построим теперь множество позиций $WhiteRook$, получающихся после хода любой белой ладьи:

$$WhiteRook = \cup_{1 \leq i \leq 64, y_i = 3} \cup_{1 \leq l \leq 4} WhiteRookMoves(i, l),$$

$$WhiteRookMoves(i, l) = \cup_{k \in RI(i,l)} WhiteRookMove(i, l, k).$$

Каждый ход ладьи выполняется в одном из 4-х направлений на длину, не превосходящую k свободных клеток, и эта длина определяется множеством RI:

$$RI(i, 2) = \{k | x_g(i) - k \geq 1\}, \qquad RI(i, 1) = \{k | 1 \leq k \leq 8 - x_v(i)\},$$

$$RI(i, 3) = \{k | x_g(i) + k \leq 8\}, \qquad RI(i, 4) = \{k | 1 \leq k < x_v(i)\}.$$

Множество $WhiteRookMove$ состоит не более чем из одной позиции:

$$WhiteRookMove(i, l, k) = \{\bar{z}|PR(i, l, k), j = fR(i, l, k), ((8 \leq y_j \leq 12) \vee$$

$$\vee(y_j = 0)), \forall\ t(((1 \leq t \leq 68) \wedge (t \notin \{i, j, 65, 67, 68\})) \to z_t = y_t),$$

$$z_i = 0, z_j = 3, z_{65} = 0, z_{67} = fWRK(i, y_{67}), z_{68} = fKBO(j, y_{68}), z_0 = 1, SafeKing(\bar{z})\}.$$

Формула PR, функция fK, а также функция $fKBO$ были введены ранее при определении формулы $SaveWKing$ и множества $WhiteKing$. Функция $fWRK$ фиксирует потерю возможности рокировки белыми в длинную или короткую стороны в случае хода белой ладьей с углового поля:

$$fWRK(i, y_{67}) \rightleftharpoons \begin{cases} 2 \cdot \lfloor y_{67}/2 \rfloor, & \text{при } i = 1, \\ y_{67} - 2 \cdot \lfloor y_{67}/2 \rfloor, & \text{при } i = 8, \\ y_{67}, & \text{при } i \notin \{1, 8\}. \end{cases}$$

Множество позиций $WhiteBishop$, получающихся после хода любого белого слона, записывается аналогично множеству $WhiteRook$ с заменой вертикальных и горизонтальных направлений диагональными:

$$WhiteBishop = \cup_{1 \leq i \leq 64, y_i = 4} \cup_{1 \leq l \leq 4} WhiteBishopMoves(i, l),$$

$$WhiteBishopMoves(i, l) = \cup_{k \in BI(i,l)} WhiteBishopMove(i, l, k),$$

$$BI(i,1) = \{k|1 \le k \le 8 - x_v(i), x_g(i) + k \le 8\},$$
$$BI(i,2) = \{k|1 \le k \le 8 - x_v(i), x_g(i) - k \ge 1\},$$
$$BI(i,3) = \{k|1 \le k < x_v(i), x_g(i) + k \le 8\},$$
$$BI(i,4) = \{k|1 \le k < x_v(i), x_g(i) - k \ge 1\},$$

$$WhiteBishopMove(i,l,k) = \{\bar{z}|PB(i,l,k), j = fB(i,l,k), ((8 \le y_j \le 12)\vee$$
$$\vee(y_j = 0)), \forall\ t(((1 \le t \le 68) \wedge (t \notin \{i,j,65,68\})) \to z_t = y_t), z_i = 0,$$
$$z_j = 4, z_{65} = 0, z_{68} = fKBO(j, y_{68}), z_0 = 1, SafeKing(\bar{z})\}.$$

Множество позиций $WhiteQueen$, получаемых после хода (любого) белого ферзя, является объединением ходов ферзя по вертикали и горизонтали, т.е. ходом ладьи, и ходов ферзя по диагонали, т.е. ходом слона:

$$WhiteQueen = \cup_{1\le i\le 64, y_i=2}(\cup_{1\le l\le 4} WhiteQBMoves(i,l) \cup_{1\le l\le 4} WhiteQRMoves(i,l)),$$

$$WhiteQBMoves(i,l) = \cup_{k\in BI(i,l)} WhiteQBMove(i,l,k),$$

$$WhiteQRMoves(i,l) = \cup_{k\in RI(i,l)} WhiteQRMove(i,l,k),$$

$$WhiteQBMove(i,l,k) = \{\bar{z}|PB(i,l,k), j = fB(i,l,k), ((8 \le y_j \le 12)\vee$$
$$\vee(y_j = 0)), \forall\ t(((1 \le t \le 68) \wedge (t \notin \{i,j,65,68\})) \to z_t = y_t), z_i = 0,$$
$$z_j = 2, z_{65} = 0, z_{68} = fKBO(j, y_{68}), z_0 = 1, SafeKing(\bar{z})\},$$

$$WhiteQRMove(i,l,k) = \{\bar{z}|PR(i,l,k), j = fR(i,l,k), ((8 \le y_j \le 12)\vee$$
$$\vee(y_j = 0)), \forall\ t(((1 \le t \le 68) \wedge (t \notin \{i,j,65,68\})) \to z_t = y_t), z_i = 0,$$
$$z_j = 2, z_{65} = 0, z_{68} = fKBO(j, y_{68}), z_0 = 1, SafeKing(\bar{z})\}.$$

Множество позиций $WhiteKNight$, получаемых после хода любого белого коня, строится аналогично множеству $WhiteKing$ ходов белого короля за вычетом хода-рокировки:

$$WhiteKNight = \cup_{1\le i\le 64, y_i=5} \cup_{1\le l\le 8} WhiteKNightMove(i,l),$$

$$WhiteKNightMove(i,l) = \{\bar{z}|PN(i,l), j = fN(i,l), ((8 \le y_j \le 12)\vee$$
$$\vee(y_j = 0)), \forall\ t(((1 \le t \le 68) \wedge (t \notin \{i,j,65,68\})) \to z_t = y_t), z_i = 0,$$
$$z_j = 5, z_{65} = 0, z_{68} = fKBO(j, y_{68}), z_0 = 1, SafeKing(\bar{z})\}.$$

Множество позиций $WhitePawn$ после ходов белых пешек состоит из множества обычных ходов белых пешек на одно поле (включая взятие фигуры) — $WhitePawnMove1$, множества ходов белых пешек со 2-й горизонтали на два поля — $WhitePawnMove2$, множества специальных ходов взятия на проходе — $WhitePawnMoveEP$ и множества ходов-превращений (в том числе путём взятия фигуры) пешек на 8-й горизонтали в другую фигуру — $WhitePawnMove8$:

$$WhitePawn = \cup_{1\le i\le 64, y_i=6}(\cup_{1\le l\le 3} WhitePawnMove1(i,l) \cup WhitePawnMove2(i) \cup$$
$$\cup(\cup_{1\le l\le 2} WhitePawnMoveEP(i,l)) \cup (\cup_{1\le l_1\le 3} \cup_{1\le l_2\le 4} WhitePawnMove8(i,l_1,l_2))).$$

Определим последовательно эти множества:

$$WhitePawnMove1(i,l) = \{\bar{z}|PWP1(i,l), j = fWP(i,l), \forall\ t(((1 \le t \le 68)\wedge$$
$$\wedge(t \notin \{i,j,65\})) \to z_t = y_t), z_i = 0, z_j = 6, z_{65} = 0, z_0 = 1, SafeWKing(\bar{z})\}.$$

Формула $PWP1$ проверяет возможность продвижения белой пешки на одно поле или взятия фигуры, а функция fWP даёт новое поле для данной белой пешки:

$$\begin{aligned}&PWP1(i,1) \rightleftharpoons ((x_v(i) \geq 2) \wedge (x_g(i) \leq 6) \wedge (8 \leq y_{i+7} \leq 12)),\\&PWP1(i,2) \rightleftharpoons ((x_v(i) \leq 7) \wedge (x_g(i) \leq 6) \wedge (8 \leq y_{i+9} \leq 12)),\\&PWP1(i,3) \rightleftharpoons ((x_g(i) \leq 6) \wedge (y_{i+8} = 0)),\\&fWP(i,1) \rightleftharpoons i+7, \quad fWP(i,2) \rightleftharpoons i+9, \quad fWP(i,3) \rightleftharpoons i+8,\end{aligned}$$

$$WhitePawnMove2(i) = \{\bar{z} | PWP2(i,l), j = i+16, \forall\, t(((1 \leq t \leq 68) \wedge$$

$$\wedge (t \notin \{i,j,65\})) \rightarrow z_t = y_t), z_i = 0, z_j = 6, z_{65} = x_v(i), z_0 = 1, SafeWKing(\bar{z})\}.$$

При ходе белой пешки на два поля очевидным образом изменяется значение переменной y_{65}, что даёт возможность чёрным на следующем ходу взять эту пешку на проходе. Формула $PWP2$ проверяет возможность хода белой пешки на два поля:

$PWP2(i) \rightleftharpoons ((x_g(i) = 2) \wedge (y_{i+8} = 0) \wedge (y_{i+16} = 0))$.

При ходе пешкой «взятие на проходе»:

$$WhitePawnMoveEP(i,l) = \{\bar{z} | PWPEP(i,l), j = fWP(i,l), \forall\, t(((1 \leq t \leq 68) \wedge$$

$$\wedge (t \notin \{i, j-8, j, 65\})) \rightarrow z_t = y_t), z_i = 0, z_{j-8} = 0, z_j = 6, z_{65} = 0, z_0 = 1, SafeWKing(\bar{z})\}$$

основной задачей является проверка возможности этого хода с помощью формулы $PWEP$:

$$PWPEP(i,1) \rightleftharpoons ((x_v(i) \geq 2) \wedge (x_g(i) = 5) \wedge (y_{i-1} = 12) \wedge (y_{i+7} = 0) \wedge (y_{66} = x_v(i) - 1)),$$

$$PWPEP(i,2) \rightleftharpoons ((x_v(i) \leq 7) \wedge (x_g(i) = 5) \wedge (y_{i-1} = 12) \wedge (y_{i+9} = 0) \wedge (y_{66} = x_v(i) + 1)).$$

Наконец, ход-превращение белой пешки, характеризующий позиции из множества $WhitePawn$ $Move8$, —

$$WhitePawnMove8(i, l_1, l_2) = \{\bar{z} | PWP8(i, l_1), j = fWP(i, l_1), \forall\, t(((1 \leq t \leq 68) \wedge$$

$$\wedge (t \notin \{i, j, 65, 68\})) \rightarrow z_t = y_t), z_i = 0, z_j = l_2 + 1, z_{65} = 0, z_{68} = fKBO(j, y_{68}), z_0 = 1, SafeWKing(\bar{z})\}$$

— является комбинацией 3-х вариантов хода пешкой (2-х вариантов взятия и обычного хода на одно поле) и 4-х вариантов её превращения в одну из белых фигур. Формула $PWP8$, проверяющая возможность этого хода, строится аналогично формуле $PWP1$:

$$\begin{aligned}&PWP8(i,1) \rightleftharpoons ((x_v(i) \geq 2) \wedge (x_g(i) = 7) \wedge (8 \leq y_{i+7} \leq 11)),\\&PWP8(i,2) \rightleftharpoons ((x_v(i) \leq 7) \wedge (x_g(i) = 7) \wedge (8 \leq y_{i+9} \leq 11)),\\&PWP8(i,3) \rightleftharpoons ((x_g(i) = 7) \wedge (y_{i+8} = 0)).\end{aligned}$$

Поскольку ходом-превращением пешки белые, вообще говоря, могут взять чёрную ладью, находящуюся на угловом поле, при четвёртом типе ходов пешек значение 68-й переменной может, вообще говоря, измениться — чёрные потеряют возможность рокировки в соответствующую сторону.

Множество $WhiteMoves(\overline{y})$ полностью определено. Множество $BlackMoves(\overline{y})$, состоящее из всех позиций, получающихся после всех допустимых ходов чёрных фигур, строится аналогичным образом — при $p_0 = 1$, т.е. при $y_0 = 1$.

Построим теперь формулы, содержащие новые позиции, получаемые из множеств $WhiteMoves(\overline{y})$ и $BlackMoves(\overline{y})$. Переменную y_{269} (y_{270}) будем рассматривать как индикатор непустоты множества $WhiteMoves(\overline{y})$ (множества $BlackMoves(\overline{y})$). Если множество $WhiteMoves(\overline{y})$ непусто, то положим

$$WM_1(\overline{y}) \rightleftharpoons ((\overline{P}(\overline{y}) \wedge (\bigvee_{\overline{z} \in WhiteMoves(\overline{y})} \forall \bigcirc \overline{P}(\overline{z}))) \wedge p_{269}),$$

$$WM_2(\overline{y}) \rightleftharpoons ((\overline{P}(\overline{y}) \wedge ((\bigwedge_{\overline{z} \in WhiteMoves(\overline{y})} \exists \bigcirc \overline{P}(\overline{z})) \wedge \forall \bigcirc (\bigvee_{\overline{z} \in WhiteMoves(\overline{y})} \overline{P}(\overline{z})))) \wedge p_{269}),$$

а иначе —

$$WM_1(\overline{y}) = WM_2(\overline{y}) \rightleftharpoons ((\overline{P}(\overline{y}) \wedge \forall \bigcirc \overline{P}(\overline{y})) \wedge \neg p_{269}).$$

Аналогичным образом, если множество $BlackMoves(\overline{y})$ непусто, то положим

$$BM_1(\overline{y}) \rightleftharpoons ((\overline{P}(\overline{y}) \wedge ((\bigwedge_{\overline{z} \in BlackMoves(\overline{y})} \exists \bigcirc \overline{P}(\overline{z})) \wedge \forall \bigcirc (\bigvee_{\overline{z} \in BlackMoves(\overline{y})} \overline{P}(\overline{z})))) \wedge p_{270}),$$

$$BM_2(\overline{y}) \rightleftharpoons ((\overline{P}(\overline{y}) \wedge (\bigvee_{\overline{z} \in BlackMoves(\overline{y})} \forall \bigcirc \overline{P}(\overline{z}))) \wedge p_{270}),$$

а иначе —

$$BM_1(\overline{y}) = BM_2(\overline{y}) \rightleftharpoons ((\overline{P}(\overline{y}) \wedge \forall \bigcirc \overline{P}(\overline{y})) \wedge \neg p_{270}).$$

Тогда формулы $Rules_1$–$Rules_4$ таковы:

$$Rules_1 = Rules_2 \rightleftharpoons \forall \Box((\neg p_0 \wedge (\bigvee_{\overline{y} \in LW} WM_1(\overline{y}))) \vee (p_0 \wedge (\bigvee_{\overline{y} \in BW} BM_1(\overline{y})))),$$

$$Rules_3 = Rules_4 \rightleftharpoons \forall \Box((\neg p_0 \wedge (\bigvee_{\overline{y} \in LW} WM_2(\overline{y}))) \vee (p_0 \wedge (\bigvee_{\overline{y} \in BW} BM_2(\overline{y})))),$$

Обозначим через $Leg(\overline{y}^0)$ множество, состоящее из позиции $\overline{y}^0$ и всех легальных позиций, полученных из $\overline{y}^0$. Тогда положим

$$LW = \{\overline{y} | y_0 = 0, \overline{y} \in Leg(\overline{y}^0)\}, \qquad LB = \{\overline{y} | y_0 = 1, \overline{y} \in Leg(\overline{y}^0)\}.$$

Поскольку исходная позиция $\overline{y}^0$ входит в формулу $Init$, а все последующие легальные позиции получаются по формулам $WM_1(\overline{y})$, $WM_2(\overline{y})$, $BM_1(\overline{y})$ и $BM_2(\overline{y})$, множества LW, LB и, следовательно, формулы $Rules_1$–$Rules_4$ полностью определены.

4. ЗАПИСЬ ЦЕЛЕЙ. ЗАВЕРШЕНИЕ ПОСТРОЕНИЯ ФОРМУЛ

В настоящем разделе завершается построение формул Θ_1–Θ_4 — строятся четыре основные цели: Aim_1 — белые могут выиграть; Aim_2 — белые могут сделать ничью, но не могут выиграть; Aim_3 — чёрные могут выиграть; Aim_4 — чёрные могут сделать ничью, но не могут выиграть:

$$Aim_1 \rightleftharpoons \forall \Diamond(((p_0 \wedge \overline{P}(\overline{y})) \wedge \neg SafeBKing(\overline{y})) \wedge \neg p_{270}),$$

$$Aim_2 \rightleftharpoons ((\forall \Box(p_0 \vee ((\neg p_0 \wedge \overline{P}(\overline{y})) \wedge (p_{269} \vee SafeWKing(\overline{y})))) \wedge \forall \Box \exists \bigcirc \top) \wedge \neg Aim_1),$$

$$Aim_3 \rightleftharpoons \forall \Diamond(((\neg p_0 \wedge \overline{P}(\overline{y})) \wedge \neg SafeWKing(\overline{y})) \wedge \neg p_{269}),$$

$$Aim_4 \rightleftharpoons ((\forall \Box(\neg p_0 \vee ((p_0 \wedge \overline{P}(\overline{y})) \wedge (p_{269} \vee SafeBKing(\overline{y})))) \wedge \forall \Box \exists \bigcirc \top) \wedge \neg Aim_3).$$

Формула Aim_1 утверждает, что при любой игре чёрных получится такая позиция, в которой ход чёрных, их король атакован, и у чёрных нет защиты (допустимых ходов), т.е. чёрным — мат. Формула Aim_2 утверждает, что белые всегда (при любой игре чёрных) могут избежать мата своему королю, но не всегда (не при любой игре чёрных) могут выиграть — сами поставить мат. Такое описание цели Aim_2 в силу семантики логики ветвящегося времени и построения формулы $Rules_2$ включает в себя случай пата белым, а также троекратного повторения позиции. Смысл формул Aim_3 и Aim_4 аналогичен смыслу формул Aim_1 и Aim_2 соответственно с заменой белого цвета на чёрный. Определим теперь формулы Θ_1–Θ_4:

$$\Theta_1 \rightleftharpoons ((Init \wedge Rules_1) \wedge Aim_1),$$

$$\Theta_2 \rightleftharpoons ((Init \wedge Rules_2) \wedge Aim_2),$$

$$\Theta_3 \rightleftharpoons ((Init \wedge Rules_3) \wedge Aim_3),$$

$$\Theta_4 \rightleftharpoons ((Init \wedge Rules_4) \wedge Aim_4).$$

Построение формул Θ_1–Θ_4 завершено.

5. РАСПОЗНАВАНИЕ ВЫПОЛНИМОСТИ ПОСТРОЕННЫХ ФОРМУЛ. ПОЛУЧЕНИЕ РЕШЕНИЯ И ЕГО ОБОСНОВАНИЕ

Распознавание выполнимости построенных формул л.в.в. Θ_1–Θ_4 для данной задачи удобно проводить табличным алгоритмом распознавания выполнимости формул л.в.в., детально описанным в [8]. В этой работе доказаны теоремы о корректности и полноте данного алгоритма: если алгоритм дал ответ, что формула Θ выполнима и построил модель M, то формула Θ действительно выполнима и истинна в модели M; если формула Θ выполнима, то алгоритм даст ответ, что формула Θ выполнима и построит некоторую модель M, в которой эта формула истинна. Для формул Θ_1–Θ_4 имеет место следующая теорема.

Теорема. *Формула Θ_1 (Θ_3) выполнима тогда и только тогда, когда в исходной позиции белые (чёрные) могут выиграть. Формула Θ_2 (Θ_4) выполнима тогда и только тогда, когда в исходной позиции белые (чёрные) могут сделать ничью, но не могут выиграть.*

В случае выполнимости формулы Θ_1, либо Θ_2 (формулы Θ_3, либо Θ_4), по модели для неё получается стратегия белых (чёрных) — первый ход и последующие ходы на ответы чёрных (белых), при которой белые (чёрные) побеждают, либо делают ничью. Каждая вершина в модели соотвествует шахматной позиции, а дуга из одной вершины в другую — сделанному ходу. Примеры стратегий, получаемых по моделям для формул, содержатся в [1].

Работа выполнена при финансовой поддержке программы фундаментальных исследований Отделения математических наук РАН секция «Алгебраические и комбинаторные методы математической кибернетики» (проект «Задачи оптимального синтеза управляющих систем»).

Библиографический список

1. *Хелемендик, Р.В.* Элементы математической логики и возможности ее приложений / Р.В. Хелемендик. М.: МАТИ. 2009. с. 124.
2. http://rybkachess.com
3. *Капабланка, Х.Р.* Учебник шахматной игры / пер. с англ.; 2-е изд.; под ред. с предисл. и коммент. М.М. Ботвинника / Х.Р. Капабланка. М.: Физкультура и спорт. 1995. 151 с.
4. *Твердохлебов, В.А.* Геометрические образы законов функционирования автоматов / В.А. Твердохлебов. Саратов: Науч. книга, 2008. 183 с.
5. *Emerson, E.A.* Automated temporal reasoning about reactive systems / E.A. Emerson // Logics for concurrency. Lecture Notes in Computer Science. Berlin: Springer, 1996. Vol. 1043. P. 41–101.
6. *Emerson, E.A.* Decision Procedures and Expressiveness in the Temporal Logic of Branching Time / E.A. Emerson, J.I. Halpern // J. of Computer and System Sciences. Feb. 1985. Vol. 30, № 1. P. 1–24.
7. *Хелемендик, Р.В.* О единой формальной записи всех допустимых ходов в любой шахматной позиции / Р.В. Хелемендик // Синтез и сложность управляющих систем: материалы XVI Междунар. шк.-семинара (Санкт-Петербург, 26–30 июня 2006 г.) / под ред. О.Б. Лупанова. М.: Изд-во мех.-мат. ф-та Моск. ун-та, 2006. С. 108–112.
8. *Хелемендик, Р.В.* Алгоритм распознавания формул логики ветвящегося времени и эффективный алгоритм построения выводов общезначимых формул из аксиом / Р.В. Хелемендик // Математические вопросы кибернетики: сб. ст. / под ред. О.Б. Лупанова. М.: Физматлит, 2006. Вып. 15. С. 217–266.