

## НАВИГАЦИЯ ПО ШАШЕЧНЫМ ПАРТИЯМ В БАЗЕ ДАННЫХ ПАРТИЙ

А.И. Куев

*Адыгейский республиканский институт гуманитарных исследований, г. Майкоп*

Рассматривается информационная модель представления адреса шашечной партии в энциклопедической базе данных партий с целью эффективного поиска необходимой партии. Предлагаемая модель адресации шашечной партии может найти применение в построении новых информационно-поисковых систем.

Современное состояние цифровой экономики (ЦЭ) требует научного осмысления этого нового понятия, под которым одни понимают развитие искусственного интеллекта (ИИ) [1], другие – внедрение современных информационных технологий в экономические расчеты [2], третьи – «BigData» (большие данные) [3] или дополняют реальность жизни виртуальной частью [4]. Обобщая определения ЦЭ, можно выделить главное – цифровая экономика охватывает все то, что поддается формализации, то есть, превращению в логические схемы, а жизнь сама найдет возможность вписать это «не-что» в систему производства, распределения, обмена и потребления.

В этом контексте «цифровизацию» можно с успехом также отнести и к другим сферам, например к играм в шашки и шахматы. Одна из проблем в этих играх возникает при анализе партий. Для этого пишутся книги энциклопедического свойства, где представлена база данных с анализом определенных партий. При этом проблема состоит в поиске необходимой партии из достаточно большого объема партий и, главное трудно построить классификацию по признакам партий, за исключением, разве что, фрагментов – дебютов, эндшпилей и т.п. Если в информационных базах и информационно-поисковых системах достаточно точно формулируются записи и поиск может быть осуществлен по любым полям, то в базе шашечных партий это несколько проблематично сделать.

Предлагаемая в данной работе концепция адресации партии позволяет в цифровом формате присвоить определенный адрес любой партии по шагам, которые имели место в шашечной игре. Это весьма важное обстоятельство создает возможности эффективного ориентира в пространстве партий для аналитика. Такой подход отвечает требованиям и принципам цифровой экономики.

Опишем алгоритм выбора конкретной шашечной партии из определенной базы данных.

Адрес каждой партии может быть представлен тремя способами. В первом случае адрес представляется некоторой матрицей. Во втором – соответствующим элементом во множестве натуральных чисел, в третьем – определенным сочетанием множества натуральных чисел.

Следует отметить, что данная процедура использовалась нами для систематизации партий с целью исследования разработанной концепции «с5f4». Идея кодировки партий появилась в процессе разработки концепции «с5f4» [5].

Задача заключается не в разработке определенной части энциклопедии данных, а в несколько иной цели. Это разработка продолжения какого-то начала. В данном случае оно связано с занятием полей: с5, f4. В этом смысле наш план носит узкий, но вполне конкретный характер, подчиняющийся поставленной цели согласования с нашей концепцией. Аналитикам, которые планируют дальнейшие исследования можно в этом плане «нанизывать» на эту схему и располагать рассматриваемые партии в нужном месте.

Процесс формализации для цифровизации анализа шашечных партий можно осуществить двумя способами. По первому способу цифровизации анализа партий компьютер фиксирует возможные ходы, любые, даже самые неприемлемые.

По второму способу анализ одной партии разбивается на два уровня. На первом уровне формируются варианты, продолжения, разветвления, возможности. На втором уровне они повторяются. Начало уровня начинается с подсистем. Далее повторяются варианты продолжения, разветвления, возможности. Далее анализ продолжается: план, ход, направление, вектор, компоненты.

Систематизация анализируемых партии осуществляется следующим образом. Любой осуществляемый ход имеет свою приоритетность, которая определяется схемой «выше-направо». Приоритетность исходит из двух векторов развития шашек. Поэтому, в начале игры, первый приоритет определяется по буквам А,В,С,Д,Е,Ф,Г,Н. Далее определяется уровень второго приоритета по цифрам 1,2,3,4,5,6,7,8. На следующем этапе определяется взаимосвязь этих приоритетов. Таким образом, самым приоритетным оказывается первый элемент первого столбца а1. Далее за этим – следующий элемент – это а3 и т.д. Затем начинается рассмотрение второго столбца b2,b4,b6,b8. Например, если а1 не имеет возможности хода, то рассматривается а3. Такая процедура не зависит от цвета шашек. Эта организация расположения анализа партии легко позволяет квалифицированному шашкисту определить место любой известной и неизвестной (не сыгранной) партии в общей схеме (базы данных). Кроме этого она позволяет найти различные варианты близкие к партии, которую планируется поместить в общую схему.

Напомним, что данная процедура была разработана и использовалась для систематизации партий с целью исследования разработанной нами концепции «с5f4».

Отметим, что взятие шашки, в некоторых случаях, не включается в предлагаемую схему разветвления. Поэтому в каждом конкретном случае взятие или бой рассматриваются как отдельные варианты или способы взятия, которые не включаются в общую схему. Это делается для того, чтобы анализ был прост и прозрачен. Исключения составляют рассмотрения известных начал к которым привыкли шашкисты.

Первый способ представления удобен тогда, когда необходимо владеть информацией о том, какие партии не заняли свое место в определенной схеме. Вторым методом представления дает возможность оценки какие начала наиболее изучены, а какие нет. Это возможно в том случае, когда мы знаем какое количество партий уже проанализировано.

Предложенная схема упорядочивания вариантов позволяет автоматизировать процесс анализа партий и является удобным и эффективным средством для пользователей, которые занимаются анализом партии. Рассмотрим одну партию, которая показывает величину поля с5. Этому полю посвящено большинство исследований. Рассмотрим первый вариант кодировки партии. На рис.1 изображена диаграмма начала игры.

#### Партия первая

1.c3-d4 b6-a5 2.d4-c5 d6xb4 3.a3xc5 c7-b6 4.e3-d4 d8-c7  
5.g3-f4 c7-d6 6.b2-a3 d6xb4 7.a3xc5 b8-c7 8.c1-b2 c7-d6  
9.b2-a3 d6xb4 10.a3xc5 f6-g5 11.d2-c3 g5xe3 12.c3-b4 a5xe5  
13.f2xa5+.

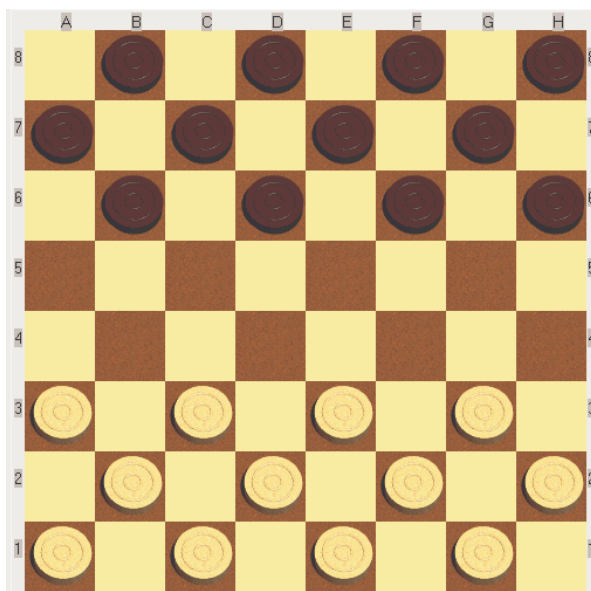


Рис.1. Диаграмма начала игры

В процессе кодирования последовательно будем заносить единичку в матрицу

$$A = \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \dots & \dots & \dots & \dots \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{bmatrix}$$

каждый элемент которой  $a_{ij} \in \{0,1\}$ .

У белых 7 вариантов ходов: 1-й: 1.a3-b4, 2-й: 2.c3-b4, 3-й: 2.c3-d4, 4-й: 12.e3-d4, 5-й: 2.e3-f4, 6-й: 2.g3-f4, 7-й: 2.g3-h4.

Обозначим каждый из них цифрами от 1 до 7. Например, 1-й: 1.a3-b4, 2-й: 1.c3-b4. Эти номера попадают в первую строку матрицы. Соответственно у черных- номера 1...7 имеют ходы 1-й: 1...b6-a5, 2-й: 1...b6-c5. Они попадают на вторую строку матрицы соответственно своим номерам. Номера соответствуют номеру элемента строки. Белые, согласно изложенной партии делают первый ход 1.c3-d4 (Рис. 2). Это отражается элементом  $a_{ij}(i=1, j=3)$  в матрице А. Сначала сформируем матрицу размерностью 1x8. Очевидно, каждый ход соответствует определенной строке матрицы. По мере движения шашек будем наращивать количество строк матрицы. Матрица в самом начале имеет вид вектора строки.

0 0 1 0 0 0 0 0

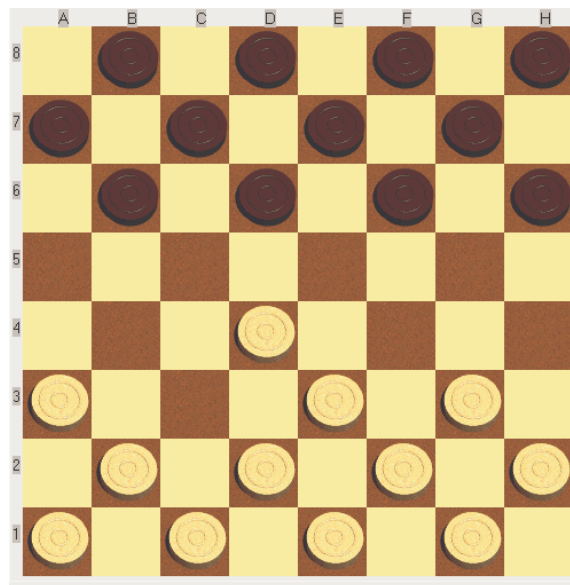


Рис 2. Диаграмма после 1.c3-d4

Далее черные имеют варианты: 1-й: 1...b6-a5, 2-й: 1...b6-c5, 3-й: 1...d6-c5, 4-й: 1...d6-e5, 5-й: 1...f6-e5, 6-й: 1...f6-g5, 7-й: 1...h6-g5. Черные выбирают первый по нумераций ход 1-й: 1...b6-a5 (Рис.3). Это соответствует первому элементу второй строки  $a_{ij}(i=2, j=1)$ . Матрица выглядит следующим образом.

0 0 1 0 0 0 0 0  
1 0 0 0 0 0 0 0

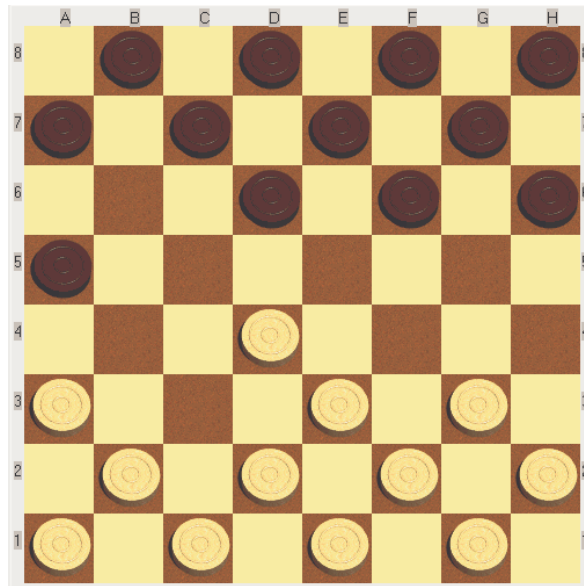


Рис. 3. Диаграмма после 1...b6-a5

Далее белые имеют варианты: 1-й: 2.a3-b4, 2-й: 2.b2-c3, 3-й: 2.d2-c3, 4-й: 2.d4-e5, 5-й: 2.d4-c5, 6-й: 2.e3-f4, 7-й: 2.g3-f4, 8-й: 2.g3-h4. Заметим, что сюда мы включаем даже глупые ходы, например, ход: 2.a3-b4. В зависимости от хода белых черные располагают также определенными вариантами ходов. Здесь следует отметить, что нумерацию следует проводить с учетом глупых ходов. Белые выбрали ход 2.d4-c5 (Рис. 4). Этот ход соответствует элементу третьей строки, более точно – 4-му элементу третьей строки. Это отражается следующим образом  $a_{ij}$  ( $i=3, j=4$ ). Матрица выглядит следующим образом.

0	0	1	0	0	0	0	0
1	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0

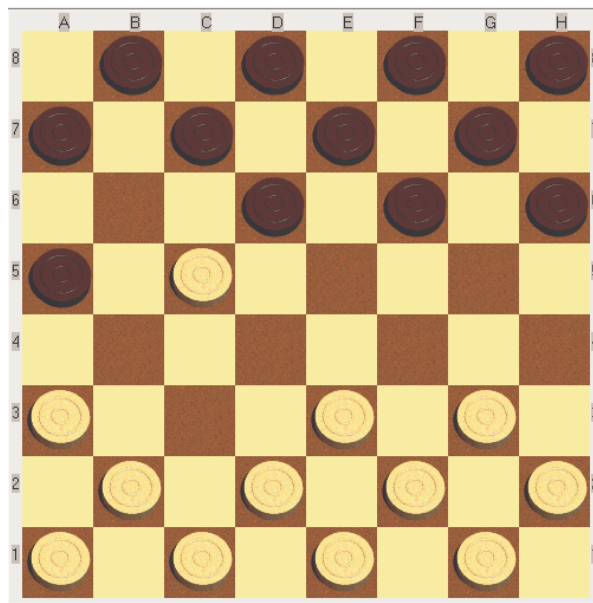


Рис. 4. Диаграмма после 2.d4-c5

Согласно этого черные обязаны сделать взятие шашки с5. Однако нам нужно пронумеровать те ходы, которые черные не могут осуществить. Это обусловлено тем, что нам необходимо сохранить предлагаемую систему. Например, нам важно, чтобы в матрице согласно алгоритму черные занимали все строки с четными номерами. С точки зрения распознавания образов можно эту ситуацию выделить, например, следующим образом. В матрице вместо нулей (поскольку они не участвуют в передаче информации) можно поставить знак «+», шашки независимо от цвета могут осуществить движение в этом направлении. Тогда в матрице символ «-» будут выражать невозможность хождения в этих направлениях. С учетом этого черные имеют варианты: 1-й: 2....a5-b4, 2-й: 2....a7-b6, 3-й: 2....c7-b6, 4-й: 2....d6xb4, 5-й: 2....d6-e5, 6-й: 2....f6-e5, 7-й: 2....f6-g5, 8-й: 2....h6-g5. Черные осуществляют взятие шашки 2....d6xb4 (Рис. 5). Это отражается в матрице заполнением элемента  $a_{ij}$  ( $i=4, j=4$ ). На диаграмме изображена позиция после взятия черными белых шашек 2....d6xb4. Матрица выглядит следующим образом

0	0	1	0	0	0	0	0
1	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0
0	0	0	1	0	0	0	0

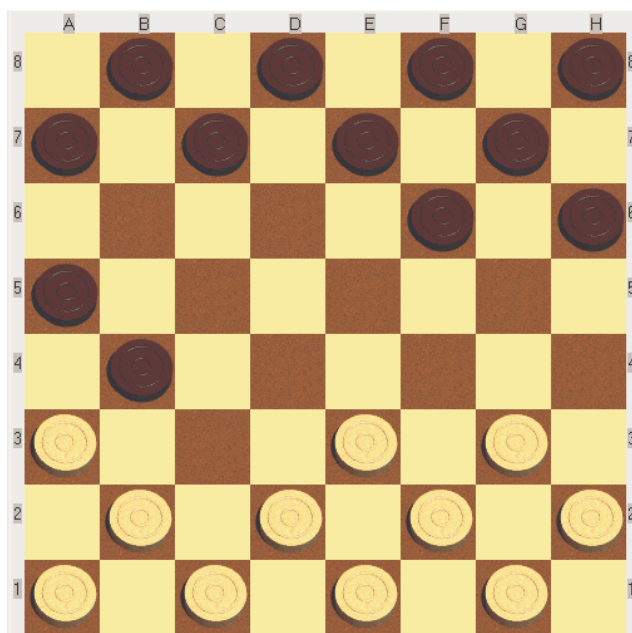


Рис.5. Диаграмма после 2....d6xb4

Далее у белых имеются следующие варианты: 1-й: 3.a3xc5, 2-й: 3.b2-c3, 3-й: 3.d2-c3, 4-й: 3.e3-d4, 5-й: 3.e3-f4, 6-й: 3.g3-f4, 7-й: 3.g3-h4. Они осуществляют взятие шашки: 1-й: 3.a3xc5 (Рис. 6). Это соответствует первому элементу в 5-й строке матрицы. Что отражается в матрице заполнением элемента  $a_{ij}$  ( $i=5, j=1$ ). Матрица выглядит следующим образом:

0	0	1	0	0	0	0	0
1	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0
0	0	0	1	0	0	0	0
0	0	0	0	1	0	0	0

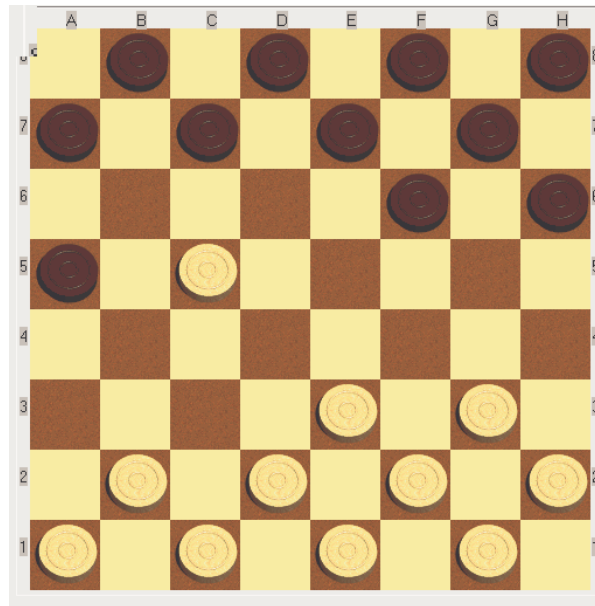


Рис. 6. Диаграмма после 3.а3х5

С учетом глупых ходов черные имеют варианты: 1-й: 3...а5-б4, 2-й: 3...а7-б6, 3-й: 3...с7-б6, 4-й: 3...с7-д6, 5-й: 3...е7-д6, 6-й: 3.ф6-е5, 7-й: 3.ф6-г5, 8-й: 3.х6-г5. Они выбирают 3-й: 3...с7-б6 (Рис. 7). Это соответствует 3-му элементу 6-ой строки матрицы. Это отражается в матрице заполнением элемента  $a_{ij}$  ( $i=6, j=3$ ). Матрица выглядит следующим образом.

0	0	1	0	0	0	0	0
1	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0
0	0	0	1	0	0	0	0
0	0	0	0	1	0	0	0
0	0	1	0	0	0	0	0

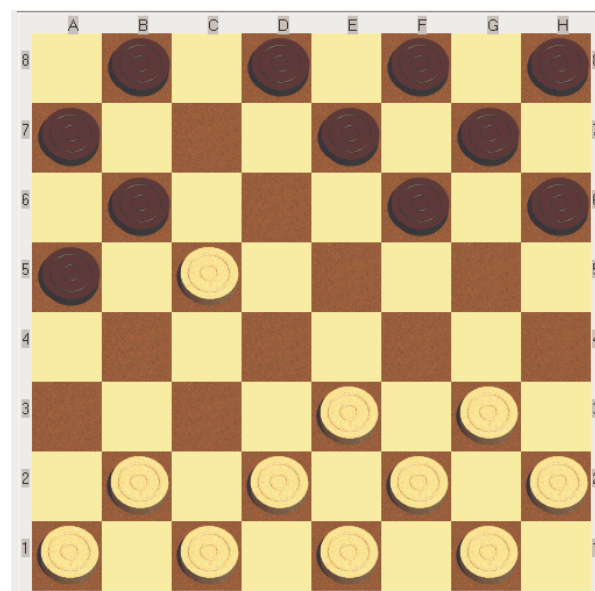


Рис. 7. Диаграмма после 3...с7-б6

Дальше расписывать процедуру не будем. Матрица заполняется аналогично в соответствии с дальнейшим развитием партии. В результате блестяще белые завершают комбинацию ходом 5-й: 12.f2xa5 и одерживают победу. Эту позицию отражает диаграмма на рис. 8. Это соответствует 2-му элементу 23-ой строки матрицы. Что отражается в матрице заполнением элемента  $a_{ij}$  ( $i=23, j=2$ ). Матрица выглядит следующим образом.

0	0	1	0	0	0	0	0
1	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0
0	0	0	1	0	0	0	0
0	0	0	0	1	0	0	0
0	0	1	0	0	0	0	0
0	0	0	0	1	0	0	0
0	0	1	0	0	0	0	0
0	0	0	0	0	0	0	1
0	1	0	0	0	0	0	0
1	0	0	0	0	0	0	0
0	0	1	0	0	0	0	0
0	1	0	0	0	0	0	0
0	1	0	0	0	0	0	0
0	1	0	0	0	0	0	0
0	1	0	0	0	0	0	0
1	0	0	0	0	0	0	0
0	1	0	0	0	0	0	0
0	1	0	0	0	0	0	0
0	0	0	1	0	0	0	0
0	0	1	0	0	0	0	0
0	0	1	0	0	0	0	0
0	1	0	0	0	0	0	0

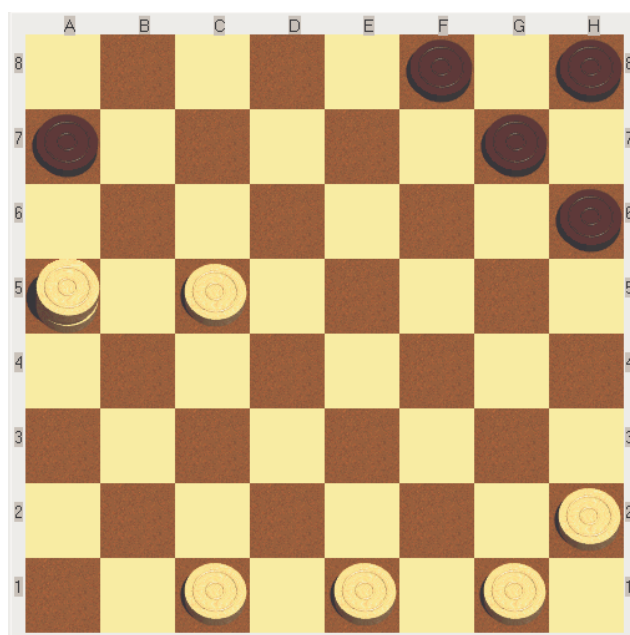


Рис.7. Диаграмма после 13.f2xa5

По результатам игры у нас получилась матрица, которая может быть представлена определенным сочетанием натуральных чисел.

Это и будет адрес рассмотренной выше партии: 3144135382132222812243321.

Такой же алгоритм может быть построен и для шахматной партии. Заметим, что эта партия не повторяется в конечном множестве всевозможных партий. При оцифровке ЭВМ запоминает партию и находит ее по этому адресу. Но нужно отметить, что позиций, которые могут повториться в различных партиях бывает довольно много. Позиции также подлежат кодировке. Кодом могут быть приняты следующие элементы: первый ход и расположение шашек. Теперь зная коды партий и позиции машина обращается к своему каталогу (базе данных) на каждом ходу. После этого машина делает вывод по какому варианту считать возможные ходы. То ли «тупо» просчитать все существующие варианты, то ли обратиться к своему опыту игры, зная результаты некоторых партий и позиций. Как это быть осуществлено, требует дополнительного исследования. Возможно здесь потребуются в некоторой степени, интуиция и опыт игрока. В данном случае нужно опираться на достижения в области нейронных сетей.

В заключение важно отметить о перспективе использования предлагаемой нами системы кодирования партии для ответа на вопрос, который многие десятилетия всех волнует. Это вопрос о том, когда же компьютер победит в шашки человека. Некоторые считают, что, «код шашек» взломан и ничего загадочного в шашечной игре не осталось. Все ходы не только записаны, но и заранее просчитаны. Поэтому сражения с компьютером больше не интересны: «железка» теперь знает, как действовать без проигрыша. С нашей точки зрения рано говорить об этом. Сегодня окончание и «живой», и «программной» шашечной партии можно прогнозировать лишь с определенной степенью вероятности. Приятным исключением на протяжении многих лет считался американский чемпион мира по шашкам Мэрион Тинсли (Marion Tinsley). С тех пор как он получил статус чемпиона мира в этой игре в 1954 году, Тинсли на официальных соревнованиях проиграл всего девять раз. С нашей точки зрения с машиной в перспективе сыграть можно только вничью (если очень постараться), либо проиграть ей. Третьего не дано.

Для того чтобы создать программу, которая позволяла бы найти оптимальную (то есть безошибочную) стратегию игры в шашки, нужно уметь четко подрезать дерево. Стратегия заключается в том, что если оба игрока будут её придерживаться, то рано или поздно придут к ничейному результату. Чтобы смоделировать усилия такой настойчивой пары, потребовалось бы проанализировать  $5 \times 10^{20}$  позиций шашек на поле. Сегодня итог игры в шашки полностью не предсказуем. Но скоро все будет предсказуемо. Вся проблема заключается не только в создании базы данных всех партий. Необ-



ходима в первую очередь суперсистема кодирования, которая позволяют быстро сравнивать позиции. А машина, как нам известно, может работать только с цифрами. Необходимо разработать таблицы соответствия позиций и цифр. Это значит, что база данных может укорочена чтобы игра была беспроигрышной программой, и тогда она будет знать все правильные ходы для любой ситуации. Имея такую мощную базу данных можно приступить к окончательному доказательству «теоремы» о ничейном исходе оптимальной шашечной игры. Надеемся, что предлагаемая нами система позволит продвинуться в решении указанных выше задач и ускорит подготовку ответа на самый важный вопрос. Прочитаны ли все ходы?

### Литература

1. Шаг в будущее: искусственный интеллект и цифровая экономика. Материалы 1-й Международной научно-практической конференции. Вып. 1 / Государственный университет управления. М.: Издательский дом ГУУ, 2017. – 289 с.
2. *Зацаринный А.А.* Информационные технологии в цифровой экономике // Проектирование будущего. Проблемы цифровой реальности: труды 1-й Международной конференции (8-9 февраля 2018 г., Москва). — М.: ИПМ им. М.В.Келдыша, 2018. – С. 29-35. URL: <http://keldysh.ru/future/2018/5.pdf>
3. *Юдина Т.Н., Гелисханов И.З.* «Экономика данных»: Big Data, цифровые платформы и цифровая рента. Сборник «Инновационные кластеры цифровой экономики: драйверы развития». Труды научно-практической конференции с международным участием. СПб: Издательство Политехнического университета, 2018. – С. 218-225.
4. *Иванов В.В., Малинецкий Г.Г.* Цифровая экономика: от теории к практике. // Инновации. – 2017. – № 12(230). – С. 3-13.
5. Куев А.И. Борьба за поле с5. – Майкоп: ООО «Качество», 2002. – 108 с.

## NAVIGATION BY CHECKERS BATCHES IN THE BATCH DATABASE

### A.I. Kuev

The information model of the representation of the address of a checked party in the encyclopedic database of parties is considered with the aim of effectively finding the necessary party. The proposed addressing model for a checker party can find application in the construction of new information search systems.