МАТЕМАТИЧКИ ОМНИБУС, 3 (2018), 103 – 115
http://im-pmf.weebly.com/matematicki-omnibus-kniga-3.html

МОДЕЛИРАЊЕ НА КОНВЕЈОВАТА ИГРА „ЖИВОТ“

Елена Атанасовска [1]

Во доцните 60-ти години, британскиот математичар Џон Хортон Конвеј ја создава играта наречена „Живот“. Името на самата игра произлегува од поврзаноста со издигнувањата, падовите и слични социјални животни состојби со кои се соочува една личност во својот живот. Тоа е симулациона игра којашто може да создаде необични шеми, со прекрасен изглед, на комплексен и интересен начин.

Еднобојни пулови се позиционираат врз бесконечна решетка. Играта започнува со почетна, произволно избрана состојба на пуловите, при што врз секое поле од решетката се применуваат три едноставни правила. Понатаму играта се одвива во вид на итерации коишто зависат само од почетната конфигурација, а не и од одлуката на играчот. Оттука, оваа игра може да се смета за игра со 0 играчи. Секое поле од решетката има 8 соседни полиња, т.е. „соседи“, или поточно кажано, 8 полиња кои делат 1 или 2 заеднички темиња со тоа поле.

На самиот почеток, играчот формира *почетен распоред* на фигури (наречени *пулови*), на полињата од една решетката. На кое поле играчот ќе смести пул, а на кое не, се препушта на самиот играч и зависи само од него. Полињата на кои има пул, односно фигура, ги нарекуваме *живи полиња,* а оние без пул се *мртви полиња.* Оваа *почетна фаза* на играта е единствениот дел од играта во кој играчот донесува каква било одлука. Веќе натаму, во *главната фаза* се применуваат одредени правила врз почетната конфигурација, со што влијанието на играчот врз последователниот развој на играта е целосно отстрането.

1. ПРАВИЛА НА ИГРАТА

Во *главната фаза* има три едноставни правила, кои се повторуваат во секоја итерација. Со нивна константна примена се создаваат нови распореди на пулови и секој од нив може да биде различен од претходните.

Прво правило. Ако моментално разгледуваното поле има точно две соседни полиња кои се живи, тогаш положбата на тоа поле (без разлика дали е живо или мртво) останува непроменета во новиот распоред. Ова правило е познато како *status quo*.

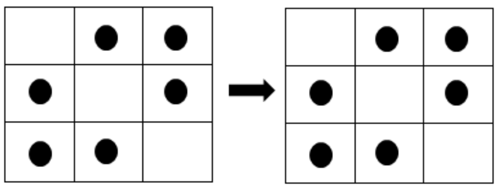
Второ правило. Ако моментално разгледуваното поле има точно три живи соседни полиња, тогаш тоа поле ќе биде живо во новосоздадениот распоред. Ова правило се нарекува *правило на раѓање* (*birth rule*).

Трето правило. Ако моментално разгледуваното поле има помалку од две или повеќе од три живи соседни полиња, тогаш тоа поле ќе биде мртво во новиот распоред. Ова правило се нарекува *правило на смрт* (*death rule*) и се дели на две потправила и тоа, *смрт од изолација* (*death by isolation*) и *смрт од пренаселеност* (*death by overcrowding*).

Почетниот распоред го означуваме со $t=0$, следниот распоред што се добива од $t=0$ со помош на трите правила го означуваме со $t=1$, итн.

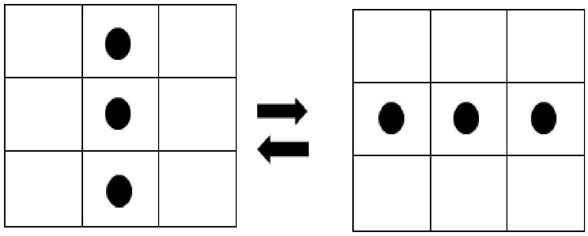
2. ПРИМЕРИ НА КАРАКТЕРИСТИЧНИ РАСПОРЕДИ

Прв карактеристичен тип на распоред е *постојаниот распоред*. Со други зборови, ова е распоред кој никогаш не се менува. Поточно, во секое време t, секое живо поле има две или три живи соседни полиња, додека секое мртво поле има помалку од три или точно шест живи соседни полиња. Значи, секое живо поле останува живо (бесмртно), а секое мртво поле останува мртво (постојано мртво) (Слика 1.). Покрај овој, постојат и многу други постојани распореди, [3].



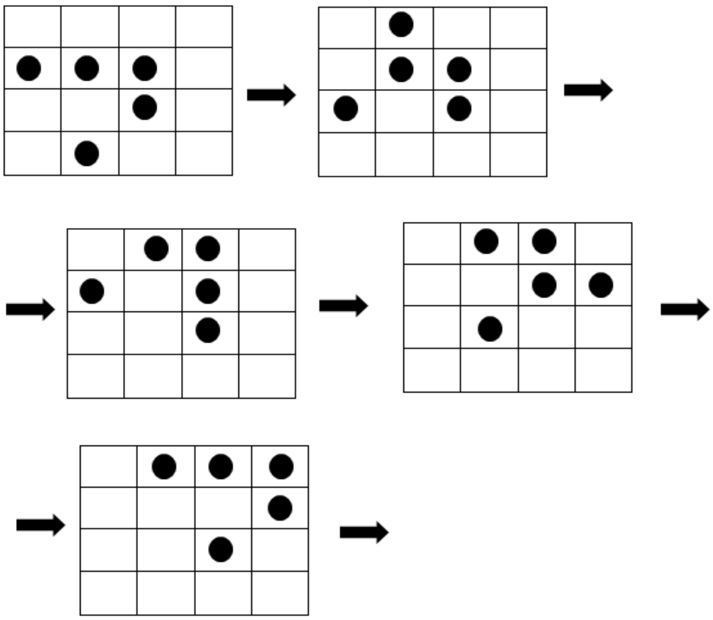
Слика 1. Постојан распоред

Вториот карактеристичен тип на распоред се нарекува *осцилатор.* Осцилатор е распоред кој постојано се повторува, со одреден период на повторување. Ова значи дека од почетниот распоред, со примена на пропишаните правила одреден број пати, се добива истиот почетен распоред на пулови од којшто сме ја започнале играта. Постојат голем број таканаречени основни осцилатори, како што се *трепкач* (*blinker*) и *жаба* (*toad*), чијшто период е 2. Но, постојат и осцилатори со поголем период, како на пример 3 (таканаречен *пулсер*), 4, 8, 14, 15, 30 итн., [4], коишто настануваат од сосема случајни почетни распореди.



Слика 2. Осцилатор (со период 2)

Третиот карактеристичен тип на распоред се нарекува *вселенски брод* (*spaceship*). За овој тип на распоред е специфично тоа што тој се повторува со одреден период, но на различна позиција. Поточно, со примена на правилата на играта врз почетниот распоред и добивање на нови распореди, по одреден број итерации го добиваме истиот почетен распоред, но придвижен (Слика 3). Постојат уште многу други вселенски бродови коишто изгледаат интересно кога се придвижуваат, [3].



Слика 3. Вселенски брод (со период 4).

3. ГУСТ ПОСТОЈАН РАСПОРЕД

Во овој дел ќе изложиме модел за наоѓање на најгуст постојан распоред, кој би можеле да го конструираме во изолирана област од бесконечната решетка, [1]. Да се потсетиме, постојан распоред е животен распоред што никогаш не се менува, т.е. секое поле е или *бесмртно* или *постојано мртво.*

За таа цел, дефинираме *густина на постојан распоред d* над некоја област R, којашто претставува однос помеѓу бројот на бесмртни полиња во постојаниот распоред и вкупниот број на полиња во разгледуваната изолирана област R,

$$d = \frac{\text{број на бесмртни полиња}}{\text{вкупен број на полиња}}\,.$$

Променливи. За да измоделираме постојан распоред во дадена област R, прво ќе дефинираме соодветна променлива. Во случајов, треба да ги означиме полињата e во R кои се бесмртни, како и оние кои се постојано мртви. Дефинираме одлучувачка променлива x_e со:

$$x_e = \begin{cases} 1, & \text{ако } e \text{ е бесмртно поле} \\ 0, & \text{ако } e \text{ е постојано мртво поле} \end{cases}\,. \tag{1}$$

Ограничувања. За секое поле e од R формираме ограничувања кои ги отсликуваат правилата на *мртво поле од изолација*, *мртво поле од пренаселеност* и правилото на *раѓање* во постојан распоред.

Дефинираме множество $N(e)$ од соседни полиња на полето e во R, при што ги добиваме следниве неравенства:

$$2x_e - \sum_{f \in N(e)} x_f \le 0 \tag{2}$$

$$3x_e + \sum_{f \in N(e)} x_f \le 6 \tag{3}$$

каде што f е кое било живо соседно поле на *e.*

Ограничувањето (2) произлегува од правилото на *мртво поле од изолација* според кое, кај постојаниот распоред, полето e со помалку од 2 бесмртни (живи) соседни полиња ќе биде постојано мртво.

Ограничувањето (3) е формирано врз основа на правилото *мртво поле од пренаселеност*, така што полињата *e* со повеќе од 3 бесмртни соседни полиња ќе бидат постојано мртви.

Причината поради која во (3), коефициентот на променливата x_e е 3 и од десната страна имаме 6 е таа што, ако полето *e* имаше повеќе од 6 живи соседи во времето *t*, тогаш најмалку еден од тие соседи ќе мора да има повеќе од 3 живи соседи во времето *t.* Тоа повлекува дека даденото поле ќе биде мртво во времето $t+1$. Со други зборови, во постојаниот распоред, постојано мртвите полиња не може да имаат повеќе од 6 бесмртни соседи.

Следните ограничувања го отсликуваат *правилото на раѓање,* според кое полињата *e* со точно 3 бесмртни соседни полиња мора исто така да бидат бесмртни. За овие ограничувања ќе претставиме два пристапи.

Првиот пристап вклучува три ограничувања и две помошни бинарни променливи a_e и b_e со кои ќе го регулираме *правилото на раѓање* за полето *e*, со кое се утврдува дека моментално разгледуваното поле *e* има точно 3 бесмртни соседи. Ограничувањата во овој случај би биле:

$$\sum_{f \in N(e)} x_f \leq 2 + 4a_e \tag{4}$$

$$\sum_{f \in N(e)} x_f \geq 4 - 4b_e \tag{5}$$

$$a_e + b_e \leq 1 + x_e \tag{6}$$

Притоа може да забележиме дека од (4) следува дека $a_e = 1$ кога полето *e* има 3 или повеќе бесмртни соседи, од (5) следува дека $b_e = 1$ кога полето *e* има 3 или помалку бесмртни соседи. Најпосле, за да имаме постојан распоред полето x_e треба да биде бесмртно и притоа да е отсликано правилото на раѓање ($x_e = 1$), т.е. тоа поле треба да е живо и да не се промени неговата состојба во следните чекори. За таа цел, збирот на неговите бесмртни соседи кои имаат вредност a_e и b_e треба да е 2. Тоа е можно само кога $a_e = 1$ и $b_e = 1$ истовремено се бесмртни, што следува од (6).

Вториот пристап користи 56 ограничувања за секое поле, но без помошни променливи. Поточно, секое од овие ограничувања претставува едно од $\binom{8}{3} = 56$ различни начини за прекршување на *правилото на раѓање*. Имено, за да се прекрши ова правило за некое поле e, тоа поле мора да биде постојано мртво и да има точно 3 бесмртни соседи. Ситуацијата е илустрирана со следниов пример (Слика 4).

●	●	●

Слика 4. Прекршување на правилото на раѓање, [1].

За да ги спречиме прекршувањата на правилото на раѓање, го формираме ограничувањето:

$$-x_e + \sum_{f \in S} x_f - \sum_{f \in N(e)-S} x_f \leq 2 \tag{7}$$

каде што S е 3-елементно подмножество од полиња од множеството $N(e)$.

Во вториот пристап ќе имаме уште едно ограничување коешто треба да осигура дека постојаниот распоред којшто го моделираме во областа R, останува во R. Ако полето $e \notin R$ има точно три соседи f, g и h коишто се од областа R, т.е. $f, g, h \in R$, тогаш мора да го формираме ограничувањето:

$$x_f + x_g + x_h \leq 2 \tag{8}$$

Поточно, ова значи дека полето e нема да има повеќе од 2 бесмртни соседи.

3.1. ЦЕЛОСНА ФОРМУЛАЦИЈА НА ПРОБЛЕМИТЕ

Од досега изведеното може да заклучиме дека, за да најдеме максимално густ постојан распоред кој го моделираме во некоја област R, треба да решиме една од следните две задачи на целобројното програмирање:

(ИП1):
$$\max \sum_{e\in R} x_e$$

при ограничувања
$$2x_e - \sum_{f\in N(e)} x_f \le 0\,,\ \forall e\in R$$
$$3x_e + \sum_{f\in N(e)} x_f \le 6,\ \forall e\in R$$
$$\sum_{f\in N(e)} x_f \le 2+4a_e,\ \forall e\in R$$
$$\sum_{f\in N(e)} x_f \ge 4-4b_e,\ \forall e\in R$$
$$a_e + b_e \le 1 + x_e\ ,\ \forall e\in R$$
$$\text{и } \mathrm{x}_e, a_e, b_e \in \{0,1\}, \forall e\in R$$

(ИП2):
$$\max \sum_{e\in R} x_e$$

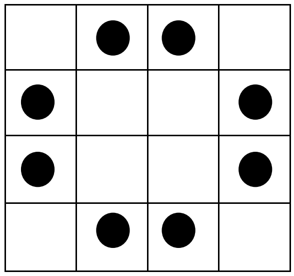
при ограничувања
$$2x_e - \sum_{f\in N(e)} x_f \le 0\,,\ \forall e\in R$$
$$3x_e + \sum_{f\in N(e)} x_f \le 6,\ \forall e\in R$$
$$-x_e + \sum_{f\in S} x_f - \sum_{f\in N(e)-S} x_f \le 2,\ \forall e\in R : \forall S\subseteq N(e) : |S| = 3$$
$$\text{така што } x_e \in \{0,1\}, \forall e\in R$$

Формулацијата (ИП2) е полесна за решавање, отколку (ИП1). Иако (ИП1) има помалку ограничувања од (ИП2), сепак бројот на променливи во (ИП2) е само една третина од бројот на променливи во (ИП1). Оттука, за решавање на (ИП2) се потребни помалку branch-and-bound јазли, помалку итерации од симплекс методот и помало CPU време на секоја од тестираните области. Исто така, најголемата област за која (ИП2) може да најде решение е од ред 9x9, додека пак (ИП1) наоѓа решение најмногу за квадратна област 7x7, [1].

3.2. НЕКОИ КОНКРЕТНИ РЕШЕНИЈА

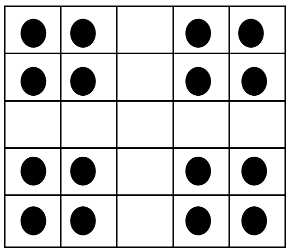
Во овој дел ќе ги изложиме решенијата на (ИП2) за квадратна $n\times n$ област R, за некои вредности на $n\in\mathbb{N}$, односно максимално густите постојани распореди над квадратни области од различни димензии. Притоа се добиваат интересни форми на постојани распореди за квадратни области од ред 4×4, 5×5, 6×6 и 9×9, [1].

Пример 1:



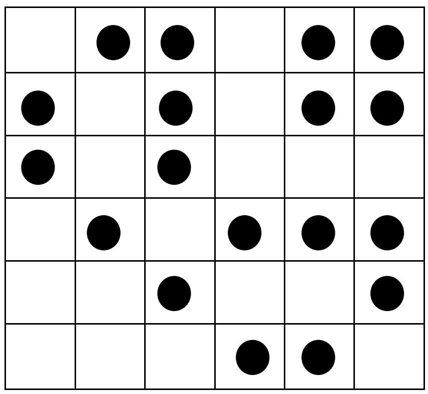
Слика 5. Максимално густ постојан распоред од ред 4×4 со густина d = 0.5.

Пример 2:



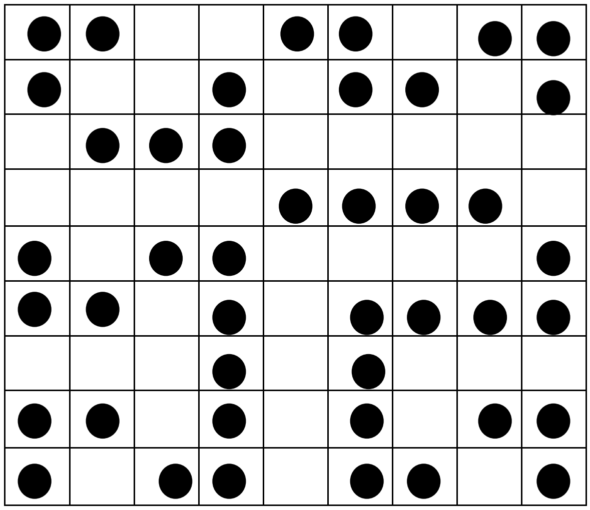
Слика 6. Максимално густ распоред од ред 5×5 со густина d= 0.64.

Пример 3:



Слика 7. Максимално густ распоред од ред 6×6 со густина d = 0.5.

Пример 4:



Слика 8. Максимално густ распоред од ред 9×9 со густина $d \approx 0.5309$.

4. МОДИФИКАЦИЈА НА НАЈГУСТ ПОСТОЈАН РАСПОРЕД

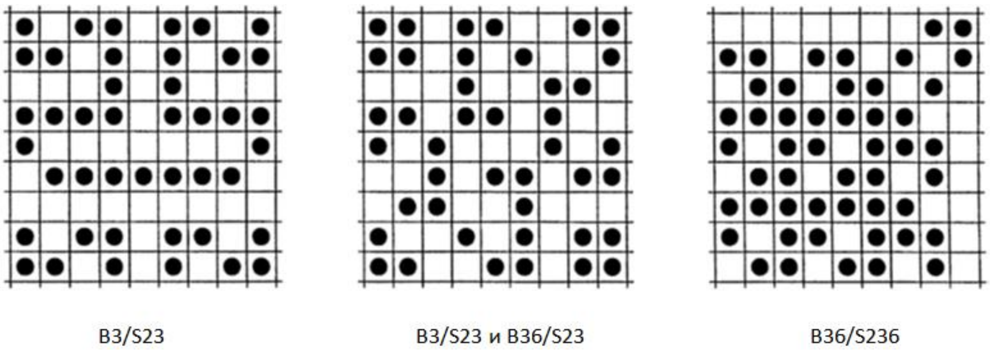
Понатаму ќе претставиме една поинаква модификација на максимално густиот постојан распоред од [1]. Поточно, во овој дел ќе претставиме нова (ИП) формулација на играта „Живот“. За разлика од претходните модели (ИП1) и (ИП2), оваа формулација е знаковно подолга и покомплицирана. Условите за поставување и почеток на модифицираната игра „Живот“ се истите што досега ги користевме, како во [1], додека промената е во правилата за раѓање, смрт и преживување. Од досега изнесениот модел знаеме дека, според правилото на:

- *преживување*, ако моментално разгледуваното поле е живо и има точно 2 или 3 живи соседи, тогаш тоа ќе остане живо и во следниот чекор (правило S23, S - survive, *преживува*);
- *раѓање,* ако моментално разгледуваното поле е мртво и има точно 3 живи соседи, тогаш тоа преминува во живо во следниот чекор од играта (правило B3, B - born, *се раѓа*);
- *умирање*, ако моментално разгледуваното поле е мртво или живо и има помалку од 2 или повеќе од 3 живи соседи, тогаш тоа поле преминува во мртво во следниот чекор.

Овој модел, што веќе го разгледавме, може да го означиме со B3/S23. Неговата модификација се состои во следниот начин на интерпретација на правилата:

– *Преживување*: ако моментално разгледуваното поле има точно 2, 3 или 6 живи соседи, тогаш тоа останува живо во следниот чекор (S236);
– *Раѓање*: ако моментално разгледуваното поле има точно 3 или точно 6 живи соседи, тогаш тоа преминува во живо (B36).

Оваа модификација на моделот B3/S23 ќе ја означиме со B36/S236. Слична модификација може да се добие доколку ги споиме правилата B36 и S23 и истата ја означуваме со B36/S23. Од чекор во чекор овие модели остануваат непроменети, т.е. добиваме постојан распоред, [2].



Слика9. Добиени постојани распореди со примена на правилата B3/S23, B3/S23 и B36/S236, B36/S236, [2].

Да претпоставиме дека ја играме играта според правилата B36/S236 и притоа бараме максимално густ постојан распоред во дадена бесконечна област *R*. Дефинираме променлива x_e исто како во [1], како и иста функција на целта. Целосната формулација би гласела:

$$\max \sum_{e \in R} x_e$$

при ограничувања

$$-x_e + \sum_{f \in S} x_f - \sum_{f \in N(e)-S} x_f \leq |S| - 1,\ \forall S \subseteq N(e) : |S| = 3 \text{ или } |S| = 6 \qquad (9)$$

$$2x_e - \sum_{f \in N(e)} x_f \leq 0 \qquad (10)$$

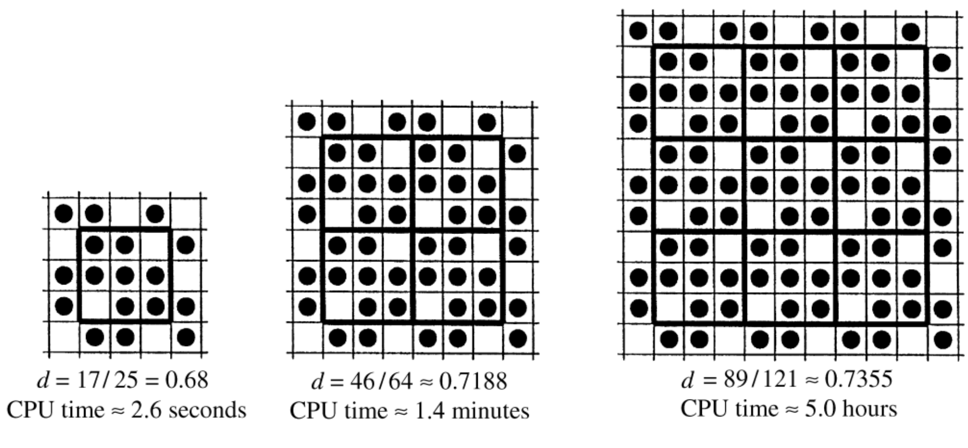
$$2x_e + 2\sum_{f \in S} x_f - \sum_{f \in N(e)-S} x_f \le 8, \forall S \subseteq N(e) : |S| = 4 \qquad (11)$$

$$2x_e + \sum_{f \in N(e)} x_f \le 8 \qquad (12)$$

$$x_f + x_g + x_h \le 2 \qquad (13)$$

Ограничувањата (9) коишто се однесуваат на правилото на раѓање, се дефинирани за триелементни или шестелементни подмножества S од соседи на *e*. Оттука, имаме $\binom{8}{3} = 56$ ограничувања кои одговараат на триелементно подмножество и $\binom{8}{6} = 28$ ограничувања кои одговараат на шестелементно подмножество. Овие правила гарантираат дека полето *e* ќе биде бесмртно ако има точно три, односно шест бесмртни соседи, соодветно. Трите ограничувања (10), (11) и (12) кои се однесуваат на правилото на смрт содржат во себе услов кој гарантира дека полето ќе биде постојано мртво ако има помалку од 2 бесмртни соседи, точно 4, точно 5 или повеќе од 6 бесмртни соседи. За точно 4 или точно 5 бесмртни соседи би имале $\binom{8}{4} = 70$ ограничувања. Крајното ограничување (13) обезбедува постојаниот распоред да остане внатре во разгледуваната област *R*, [2].

Во [2] можеме да видиме пример за најгуст постојан распоред моделиран во квадратна $n \times n$ област со модификацијата B36/S236:



Слика10. Најгуст постојан распоред на квадратна област со густина *d* и CPU времињата потребни за да се најде распоредот, [2].

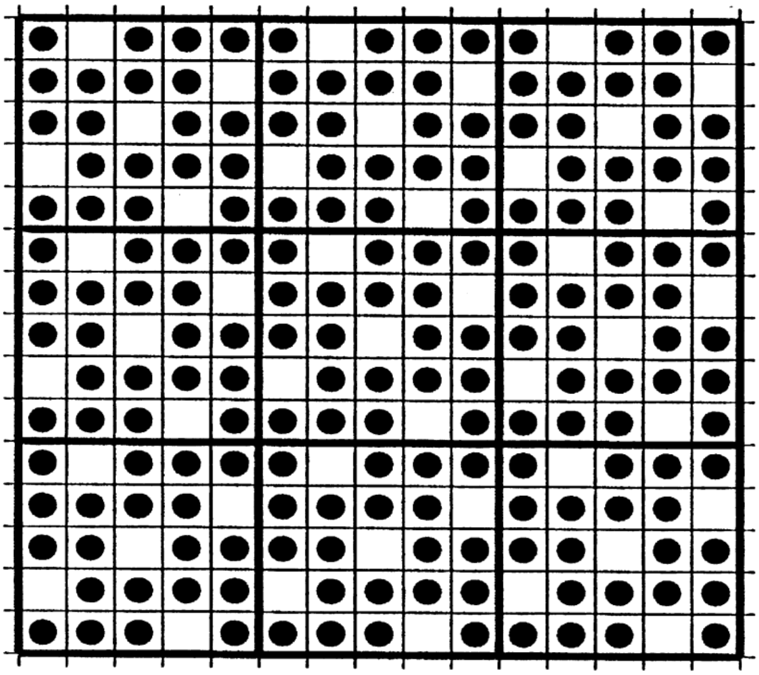
Постојаните распореди на Слика 10 се добиени со повторување на копии од 3x3 распоред за B36/S236 со густина $\frac{7}{9}$. Тој 3×3 распоред може да се искористи со повторување до бесконечност, при што конструираме бесконечен постојан распоред со густина $\frac{7}{9}$ во модифицираната игра, кој не се разликува во ништо од појдовниот распоред во конечната област 3×3. Но, оваа густина не е максимална. Најгустиот бесконечен постојан распоред е со густина $\frac{4}{5}$ (Слика 11).

За да го утврдиме максимално густиот постојан распоред во играта B36/S236, ја дефинираме следната теорема.

Теорема. ([2]) *Ако d_n е густината на најгустиот постојан распоред кој може да се конструира во квадратна nxn област R од играта B36/S236, тогаш*

$$d_n \leq \frac{4}{5}\left(1+\frac{1}{n^2}\right).$$

Доказот на теоремата може да се прочита во [2].



Слика 11. Најгустиот бесконечен постојан распоред со густина $\frac{4}{5}$ за играта B36/S236, [2].

Оваа модификација на играта е најблиска со моделот (ИП2), или поточно кажано, е еден вид на негово проширување. За нивно решавање се користат методи од целобројно програмирање коишто може да се разгледаат поподробно во [1] и во [2].

ЛИТЕРАТУРА

[1] R. A. Bosch, Integer Programming and Conway's Game of Life, SIAM Rev., 41(3), 1999, 594 – 604.

[2] Robert A. Bosch, Maximum density stable patterns in variants of Conway's game of Life, Operations Research Letters 27 (2000) 7 – 11.

[3] Eric Weisstein’s Encyclopedias, *The LifeCellular Automaton*
http://www.ericweisstein.com/encyclopedias/life/
http://www.ericweisstein.com/encyclopedias/life/topics/Spaceships.html

[4] Universität Bielefeld,
http://wwwhomes.uni-bielefeld.de/achim/freq_top_life.html

[1] Универзитет „Св. Кирил и Методиј“, Скопје
Природно-математички факултет,
ул. Архимедова 3, 1000 Скопје, Р. Македонија
e-mail: elena.atanasovska995@hotmail.com

Примен: 04. 03. 2018
Поправен: 20. 07. 2018
Одобрен: 31. 07. 2018
Објавен на интернет: 28.08.2018