

# Analogikai celluláris számítógépek – egy új paradigma a számítástechnikában

## Analogic Cellular Computers – A New Computational Paradigm

ERCSEY-RAVASZ Mária<sup>1,2</sup>, ROSKA Tamás<sup>2</sup>, NÉDA Zoltán<sup>1</sup>

1. Babeş-Bolyai Tudományegyetem, Fizika Kar, Elméleti- és számítógépes Fizika Tanszék  
str. Kogalniceanu 1, RO-400084 Cluj-Napoca,

2. Pázmány Péter Katolikus Egyetem, Információs Technológia Kar,  
Práter u. 53., Budapest, HU-1053  
ravasz@digitus.itk.ppke.hu

### ABSTRACT

*The computational paradigm represented by Cellular Neural/nonlinear Networks (CNN) and the CNN Universal Machine (CNN-UM) as an Analogic Cellular Computer, gives new perspectives for computational sciences. Beside the applications developed in image processing, robotics, we will present how this computer can be used in physics for solving special, complex problems.*

### ÖSSZEFOGLALÓ

*A celluláris neurális hálózatok (CNN) elmélete alapján megtervezett analogikai celluláris számítógépek új perspektívát nyújtanak bizonyos komplex és sok számítógépes időt igénylő feladatok megoldásában. Ezeket a chipeket eddig főképp a robotikában és képfeldolgozásban alkalmazták, de sok, fizikában fontos probléma megoldásánál is hasznosak lehetnek. A CNN alapvető elmélete és a CNN számítógép rövid bemutatása után különböző hasznos alkalmazásokról adunk áttekintést.*

**Kulcsszavak:** analogikai számítógépek, celluláris neurális hálózatok, CNN univerzális gép, Monte Carlo szimulációk, sztochasztikus folyamatok

### 1. BEVEZETÉS

A számítógépek gyors fejlődése ellenére azt tapasztaljuk, hogy sok tudományágban továbbra is komoly gondot okoz bizonyos komplex problémák megoldása vagy nagyon nagy adathalmazok kezelése. Az elmúlt évszázad során a különböző technikákkal megvalósított számítógépek teljesítménye exponenciálisan nőtt [1]. 1960 és 2000 között az egy mikroprocesszorban lévő tranzisztorok száma 1-ről közel 1 milliárdra változott. Ez alatt a négy évtized alatt egyre komplexebb processzorok jelentek meg közel azonos áron. A chipek szerveződései és működési elve azonban nem változott, máig is a Neumann János alkotta tárolt programú, digitális szerkezeti modellt követik. A mai technológia azonban lassan eléri működési korlátait, ezért ma már újfajta számítógépek konstrukciójával foglalkoznak a kutatók és mérnökök. Egy ilyen újfajta eszköz az Analogikai Celluláris Számítógép, mely a celluláris neurális hálózatok elméletére épül (rövidítve CNN-UM: Celluláris Neurális/Nemlineáris Hálózat – Univerzális Gép). A processzor működtetése a hagyományos, rugalmasan tárolt-program vezérléssel történik, azonban a műveletek jó része a folytonos analóg jelek térképszerű képein folyik.

A celluláris neurális hálózatok (CNN) elméletének [2] és a CNN univerzális gép [3] szerkezetének rövid bemutatása után áttekintést adunk a különböző alkalmazási lehetőségekről.

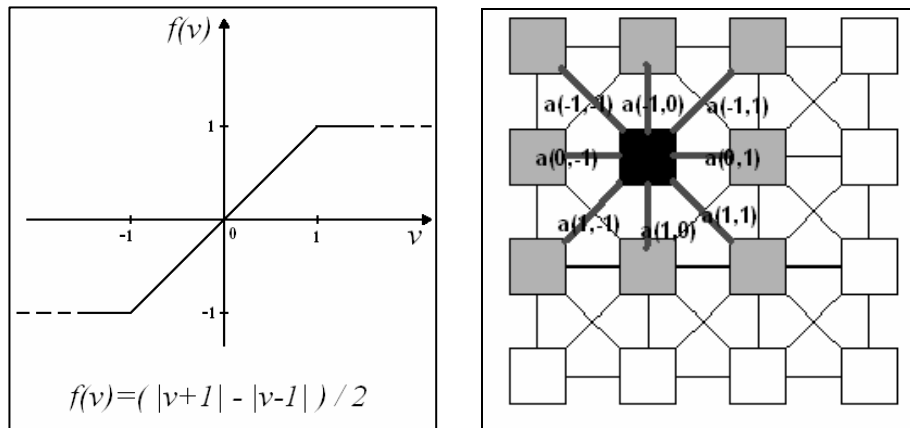
### 2. CELLULÁRIS NEURÁLIS HÁLÓZATOK

Az analogikai celluláris számítógép alapja a **celluláris neurális hálózat** vagy röviden CNN (Cellular Neural/nonlinear Network). A celluláris neurális hálózatok elméletét 1988-ban publikálták [2]. Ez egy csatolt, analóg, nem lineáris processzor-tömb, amely alapértelmezésben egy négyzet rácson elhelyezkedő, lokálisan

összekötött cella-rendszert alkot. A négyzetrács minden csúspontjában egy cella (áramkör) van. Minden cellát három érték jellemez: az  $u$  bemeneti feszültség, mely időben állandó, az  $x(t)$  állapot érték, és az  $y(t)$  kimeneti feszültség. A kimeneti feszültség egy bizonyos tartományban ( $-1$  és  $1$  között) megegyezik az állapotértékekkel, azon kívül pedig állandó értéke van, egész pontosan a következő függvénnyel írható le (1. ábra):

$$y = f(x) = \frac{1}{2}(|x+1| - |x-1|) \quad (1)$$

A továbbiakban gyakran fogjuk a celluláris hálózat állapotát képként ábrázolni, ilyenkor a fehér szín a  $-1$  kimeneti értéket, a fekete a  $+1$  kimeneti feszültséget jelöli, a köztes értékek pedig a szürke skála színei.



1. ábra

a) A kimeneti érték és állapotérték közötti összefüggést leíró függvény. b) A CNN hálózat struktúrája.

A cellák csak a szomszédokkal vannak kapcsolatban. Minden cella állapotát befolyásolhatják a szomszédok kimeneti és bemeneti értékei. A kölcsönhatások erősségét az  $A$  illetve  $B$  mátrixok határozzák meg. A cellák állapotegyenlete a következő:

$$\frac{dx_{ij}(t)}{dt} = -x_{ij}(t) + \sum_{k=-1}^1 \sum_{l=-1}^1 A_{kl} y_{i+k, j+l}(t) + \sum_{k=-1}^1 \sum_{l=-1}^1 B_{kl} u_{i+k, j+l} + z_{ij} \quad (2)$$

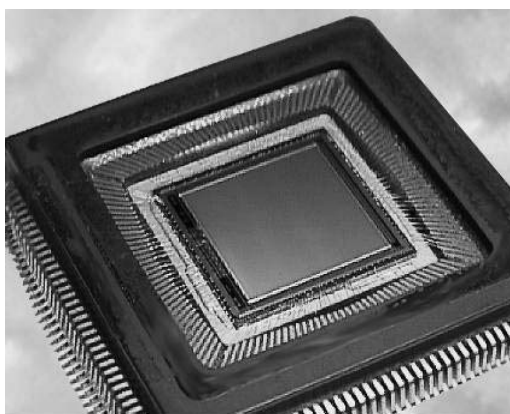
Az  $A$  és  $B$  mátrixok értékei változtathatók, de egy művelet során minden cella esetében azonosak. A  $z$  paraméter egy időben állandó érték, mely celláról cellára változhat. Az  $A, B, z$  paraméterek összességét template-nek nevezzük. Egy művelet elvégzésekor megadjuk a cellák bemeneti értékeit, definiálunk egy kezdeti állapotot, és megadjuk a template értékeit. A művelet során a cellák párhuzamosan működve és kölcsönhatva eljutnak egy stabil állapotba, amely valójában az állapotegyenletek megoldása. Ha a template-t úgy definiáltuk, hogy a kimeneti értékek végig a  $(-1, 1)$  tartományban maradnak, akkor egyetlen művelet megold egy lineáris differenciál egyenletrendszer. A kimeneti érték levágása miatt olyan template-k is definiálhatók, amelyek logikai vagy más aritmetikai műveleteknek felelnek meg.

### 3. A CNN UNIVERZÁLIS SZÁMÍTÓGÉP

Az analogikai celluláris számítógépek fizikai megvalósítása jóval később kezdődött [3,4] és még mindig fejlesztés alatt áll. Ezeket a számítógépeket CNN univerzális számítógépnek is nevezik, mert a számítógép alapját egy CNN hálózat képezi. A CNN hálózat elméletében leírt cellák lokális memóriákkal, lokális logikai és analóg egységekkel vannak ellátva, és egy globális programozó egység biztosítja, hogy a számítógép tárolt-program vezérléssel működhessen. Ezek a chipek könnyen kapcsolhatók a hagyományos számítógépekhez, melyekről különböző programozó felületek (pl. az Analogic Macro Code [5]) segítségével vezérelhető a CNN számítógép. A 2. ábrán a  $128 \times 128$  cellából álló ACE16K chip látható. A táblázatban feltüntettük az eddig gyártott chipeket. Látható, hogy a méretük egyre nő, és fontos kiemelni, hogy több rétegű chipeket is gyártanak már. Ezekben több réteg cella helyezkedik el, és a függőleges szomszédok között is állítható a kölcsönhatás erőssége. A CNN univerzális számítógépet több fontos tulajdonság teszi különlegessé [3,4]. Ezek közül a legfontosabbak: a) analóg, folytonos jelekkel dolgozik, ugyanakkor logikai műveleteket is végez (ezért nevezük analogikai számítógépnek), b) a cellák állapotértékei időben folytonosan változnak, c) térben diszkrét, d)

a cellák párhuzamosan működnek, e) univerzális Turing gép. A CNN számítógép másik fontos előnye, hogy a cellákhoz különböző (pl. fény-) érzékelők csatolhatók, ez nagyon hasznossá teszi a robotika, bionika területén is. Ezekről az alkalmazásokról röviden a 4.2. fejezetben lesz szó.

<i>Név</i>	<i>Év</i>	<i>Méret</i>
--	1993	12*12
ACE440	1995	20*22
POS48	1997	48*48
ACE4k	1998	64*64
CACE1k	2001	32*32*2
ACE16k	2002	128*128
XENON	2004	128*96*2
EYE-RIS	2005	176*144
CACE2k	Gyártás alatt	32*32*3



2. ábra

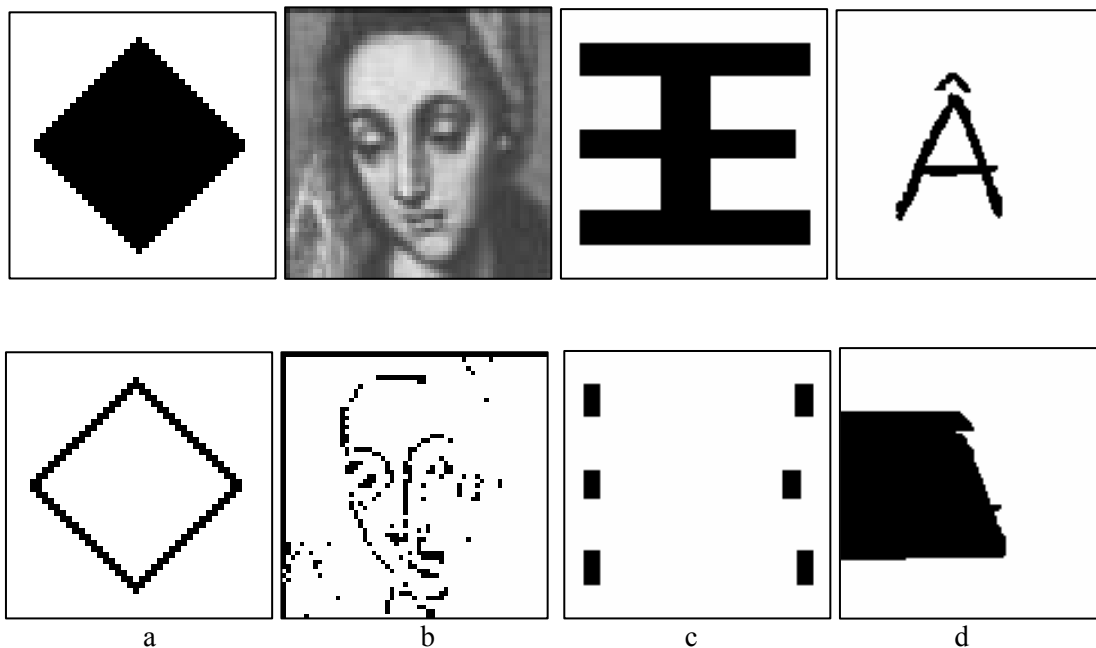
*A táblázat a CNN chippek fejlődését követi végig. A képen az ACE16K chip látható.*

#### 4. A CNN SZÁMÍTÓGÉP ALKALMAZÁSAI

A CNN számítógép struktúrája és működési elve rengeteg területen ad alkalmazási lehetőséget. Ezek közül a legfontosabbak: a) a képfeldolgozásban, b) a robotikában és bionikában; a matematika és fizika területén c) sejtautomata modellek szimulálására, d) parciális differenciál egyenletek megoldására, e) sztochasztikus szimulációk elvégzésére alkalmas.

##### 4.1. Alkalmazások a képfeldolgozásban

A celluláris neurális hálózat celláinak kimeneti értékei könnyen felfoghatók mint egy szürke skálájú kép, ezért teljesen természetes, hogy ez a CNN hálózaton dolgozó gép a képfeldolgozásban nagy szerepet játszik. Egyetlen beprogramozott művelettel az egész képen elvégezhető a transzformáció, hiszen a cellák párhuzamosan működnek. Ez a számítógép nem valós számokon, hanem képeken (2 dimenziós számtömbökön) végzi a műveleteket.



3. ábra  
Képfeldolgozási műveletek

A 3. ábrán pár egyszerű műveletet mutatunk be. A felső sorban a bemeneti képek, az alsó sorban a művelet eredménye látható. Az első template megtalálja egy fehér-fekete ábrán a széleket. A template értékei:  $A=\{0,0,0,0,1,0,0,0,0\}$ ,  $B=\{-1,-1,-1,-1,8,-1,-1,-1,-1\}$ ,  $z=-1$ . A második művelet egy szürke képen detektálja a kontúroket:  $A=\{0,0,0,0,2,0,0,0,0\}$ ,  $B=\{-1,-1,-1,-1,8,-1,-1,-1,-1\}$ ,  $z=-0.5$ . A harmadik példában a sarkokat lehet egyetlen művelettel kijelölni:  $A=\{0,0,0,0,1,0,0,0,0\}$ ,  $B=\{-1,-1,-1,-1,4,-1,-1,-1,-1\}$ ,  $z=-5$ , a negyedik template pedig árnyékot vetít:  $A=\{0,0,0,0,2,2,0,0,0\}$ ,  $B=\{0,0,0,0,2,0,0,0,0\}$ ,  $z=0$ . Ezek nagyon egyszerű példák, de természetesen több template segítségével sokkal komplexebb képfeldolgozási algoritmusok készíthetők [6,7].

#### 4.2. Alkalmazások a robotikában és bionikában

A CNN chipek fontos előnye, hogy a cellákra érzékelők is szerelhetők [8]. Pl. a Bi-i nevű gépben olyan CNN chip van, amelynek cellái fotoszenzorokat tartalmaznak, így nagyon gyors kameraként tud működni [9]. Egy másodperc alatt több ezer képet képes felvenni. Ez a szám akkor válik igazán megdöbbentővé, ha belegondolunk, hogy a mi szemünk már 50 kép/s vetítési sebességnél is folytonos képet lát. Normális videokamerák 100-nál kevesebb képet vesznek fel egy másodperc alatt. A gyorsasága mellett fontos, hogy a felvett képeken azonnal műveleteket tudunk végezni, esetleg eldönthető, hogy az adott képet érdemes-e elmenteni vagy nem (pl. ha azt akarjuk, hogy csak akkor mentsük el a képet, ha mozdult valami). Hasonlóan tapintó- és hangérezékelők tervezésén is dolgoznak a kutatók. Ezek a mesterséges érzékelők nemcsak a robotikában hasznosak, de az orvostudományban is, különböző implantátumok tervezésénél.

#### 4.3. Sejtautomata modellek

A párhuzamosan működő cellák nagyon könnyűvé teszik sejtautomata modellek programozását is. A sejtautomata modellek lényege, hogy minden cella új állapotát a lokális környezet előző (vagy még régebbi) állapota határozza meg. A sejtautomata modell szabályai általában könnyen megfogalmazhatók template-k formájában. Ezeket a műveleteket pedig egymás után ismételtethetjük, mindig az előző művelet kimeneti állapotát használva a következő bemeneti állapotaként. A párhuzamosan működő cellák nagyon gyorsá teszik ilyen modellek szimulálását.

Két alapvető példa a dilatáció és az erózió, amelyek egyetlen template-tel megoldhatók. A dilatáció esetében meghatározhatjuk, hogy minden olyan cella új állapota legyen fekete, amelynek a bizonyos szomszédai feketék voltak:  $A=\{0,0,0,0,0,-1,0,0,0,0,0\}$ ,  $B=\{0,1,0,1,1,1,0,1,0\}$ ,  $z=2$ , az erózió esetében pedig minden olyan cella legyen fehér, amelynek bizonyos szomszédai fehérek voltak:  $A=\{0\}$ ,  $B=\{0,1,0,1,1,1,0,1,0\}$ ,  $z=-2$ . Több template felírásával sokkal komplexebb, akár különböző logikai műveleteket is tartalmazó sejtautomata modellek is tanulmányozhatók [10].

#### 4.4. Parciális differenciál egyenletek megoldása

Egy template paramétereinek definiálása után, ha a celluláris neurális hálózatot elindítjuk egy kezdeti állapotból, a végső stabil állapot a rendszer állapotegyenleteinek megoldását adja. Ha a CNN számítógéppel egy konkrét differenciál egyenletet szeretnénk megoldani, térben diszkretizálva az egyenletet megtalálható a megfelelő template. Pl. ha a diffúziós egyenletet szeretnénk megoldani [11], térben diszkretizálva egy olyan lineáris differenciál egyenletet kapunk (3. egyenlet), amelynek az alakja megegyezik a cellák állapotegyenleteinek az alakjával, és ezáltal azonosíthatók a diffúziót elvégző template paramétereik.

$$\frac{\partial u(x, y, t)}{\partial t} = c \nabla^2 u(x, y, t) \Rightarrow \frac{\partial u_{i,j}}{\partial t} = \frac{c}{h^2} (u_{i,j+1} + u_{j,i+1} + u_{i,j-1} + u_{j,i-1} - 4u_{i,j}), h = \Delta x = \Delta y \Rightarrow \quad (3)$$
$$A = \{0, c/h^2, 0, c/h^2, 1 - 4c/h^2, c/h^2, 0, c/h^2, 0\}, B = \{0\}, z = 0.$$

Ezt a komplex műveletet a CNN chip kb. 20 ns alatt képes elvégezni.

Több réteg cellát tartalmazó CNN számítógépekkel sokkal bonyolultabb parciális differenciál egyenletek (PDE) oldhatók meg. Két réteg esetében pl. reakció-diffúzió típusú PDE-k [11], vagy 3 rétegű chipen már megoldható a két dimenziós Navier-Stokes egyenlet is [12].

#### 4.5. Sztochasztikus szimulációk a CNN számítógépen

A sztochasztikus szimulációk a fizikának egy másik nagyon tág témaköre, melyben fontos lenne a szimulációk gyorsítása. Itt a legnagyobb előnyt a CNN chip valós zajja jelenti, amelyet fel lehet használni valószínű véletlen-számok előállítására. A sztochasztikus vagy Monte Carlo típusú szimulációk egyik legfontosabb lépése a random számok generálása. A mai digitális számítógépeken viszont csak pszeudo-random számok generálhatók, hiszen minden algoritmus teljesen determinisztikus. A CNN számítógép analóg jelei viszont valódi – az áramkörökben lejátszódó különböző fizikai jelenségekből származó – zajt hordoznak. Ez a valós zaj azonban térben erősen korrelált, nem lehet direkt módon felhasználni véletlenszámok generálására.

Ennek a valószínű véletlenszám generátornak a működését a [13] cikkben mutattuk be, itt röviden vázoljuk. A CNN számítógépen már kidolgoztak különböző pszeudo-random generátorokat, amelyek valójában olyan kaotikus sejtautomata modellek, amelyek statisztikája jól megközelíti a véletlenszámok statisztikáját [10]. Ezek mind 1/2 valószínűséggel generálnak 0 (fehér) vagy 1 (fekete) értékeket, főleg a kriptográfiában használták őket. Mi egy ilyen kaotikus sejtautomatát használunk, csak minden lépésben megzavarjuk egy zajos képpel. A sejtautomata modell szabálya:

$$x_{t+1}(i, j) = (x_t(i+1, j) \vee x_t(i, j+1)) \oplus x_t(i-1, j) \oplus x_t(i, j-1) \oplus x_t(i, j), \quad (4)$$

ahol a  $\vee$  a vagy, a  $\oplus$  a kizáró-vagy műveletet jelenti.

A zajos képet úgy kapjuk, hogy egy  $a$  értékű szürke képen egy  $a+z$  küszöbű vágást végzünk (azok a cellák amelyeknek az értéke nagyobb mint  $a+z$  feketék lesznek, a többi fehér lesz). Ha a  $z$  egy kicsi pozitív szám, a zaj miatt mindig lesz pár fekete pixel a kimeneti képen. Nagyon korrelált zajt kapunk, de az időbeli fluktuációk valós zajból származnak. Most a zajos kép és a sejtautomata modell kimeneti képe között egy kizáró-vagy műveletet végzünk, ezzel végül is néhány véletlenszerűen választott pixel értékét megváltoztatjuk (a kizáró vagy művelet biztosítja, hogy a fehér-fekete pixelek sűrűsége nem változik meg). Ez a kis perturbáció a sejtautomata modell következő lépéseiben aztán „továbbhalad”, hiszen a szomszédok következő értékeit majd befolyásolják a megváltoztatott pixelek. Így elérhető, hogy ugyanabból a kezdeti állapotból kiindulva a random-generátor soha nem ugyanazokat az értékeket generálja, mert az eredmény mindig a zajos képtől is fog függeni, de a sejtautomata modell jó statisztikai tulajdonságai is megmaradnak (nem lesznek korrelációk). Több független 0.5 sűrűségű képből könnyen nyerhetünk más  $p$  sűrűségű képeket is (lásd [13]).

Ezt a véletlen-szám generátort az ACE16K nevű, 128\*128 cellából álló chipen teszteltük. Egy kép generálásához szükséges idő kb. 116  $\mu$ s, vagyis egyetlen véletlen-számra jutó idő 7 ns. Ezzel szemben egy 2.8 GHz-es Pentium 4 számítógép 33 ns alatt generál egy véletlen-számot. Tehát nem csak a valószínű zaj jelent előnyt, hanem az algoritmus gyorsasága is jelentősen nőtt. Ez az előny a chipek fejlődése során a cellák számának növelésével várhatóan tovább fog nőni.

Miután rendelkezésünkre áll a CNN számítógépen egy jó véletlen-szám generátor, nagyon sok különböző fizikában alkalmazott sztochasztikus (vagy Monte Carlo típusú) szimulációs módszert lehet implementálni. Itt két feladat megoldásának az alapötletét mutatjuk (be a részletek megtalálhatók a [13,14] publikációkban). Ez a két feladat a perkolációs feladat és az Ising modell.

A perkolációs feladat („site-percolation problem”) lényege a következő: adott egy fekete-fehér pontokból álló négyzetrács. Minden lépésben egy fekete pontról csak annak egy fekete szomszédjára léphetünk; kér-

dés, hogy létezik-e út a négyzetrács egyik szélétől a másikig. Ha létezik út, akkor van perkoláció. Ez egy nagyon absztrakt matematikai feladat, de nagyon sok jelenség modelljében alkalmazták már: erdőtüzek, betegségek terjedésére, folyadék terjedésére porózus anyagokban, bizonyos anyagok vezetőképességének kiszámolásához. A feladat statisztikus fizika szempontjából is érdekes, a fekete pontok sűrűségét változtatva egy másodfajú geometriai fázisátalakulás észlelhető. Megkeresni a CNN számítógéppel, hogy egy képen van-e perkoláció, nagyon izgalmas feladat, mert kiderül, hogy az a viszonylag bonyolult rekurzív algoritmus amellyel az utat keressük a digitális számítógépeken, az itt egyetlen template-tel helyettesíthető. A template értékei:  $A=\{0.5,0.5,0.5, 0.5,4,0.5,0.5,0.5\}, B=\{0,0,0,0,4,0,0,0\}$ ,  $z=3$ . A bemenet maga a kép, amelyen a perkolációt keressük, a kezdeti állapot a bemenetnek csak az első sorát tartalmazza. A művelet során egy folyam indul el, a template az első sortól kiindulva kezdi rekonstruálni a bemeneti képet, így a kimenet tartalmazni fogja az összes olyan fekete pontot, amely kötődik az első sorhoz. A művelet elvégzése után csak azt kell ellenőrizni, hogy az utolsó sorban van-e fekete pont (ez egyetlen ÉS művelettel megkapható). Természetesen a feladatot statisztikailag is ellenőriztük, az eredmények jó egyezést mutatnak a digitális gépeken nyert eredményekkel [13]. Az algoritmus gyorsasága ennél a rácsmérettel (128) még nem éri utol a digitális számítógépeket, de mivel a szükséges idő lineárisan nő a rácsmérettel, és nem a rácsméret négyzetével, mint a digitális számítógépeken, a jövőben előnyt jelenthet a CNN számítógép.

Az Ising modellen egy másodfajú fázisátalakulást mutató spinrendszert tanulmányozunk. A spinek egy négyzetrácsra helyezkednek el, mindegyik spinnek ( $s$ ) két állapota lehet -1 (fehér) és +1 (fekete), és az első szomszédok között egy ferromágneses kölcsönhatás van. A rendszer energiája:  $H = -J \sum_{\langle i,j \rangle} s_i s_j$ . Ez a köl-

cönhatás rendezné a rendszert, a spinek azonos állapotba akarnak kerülni ha  $J>0$ , de ugyanakkor a hőmérséklet zajt (rendezetlenséget) visz be a rendszerbe. Létezik egy kritikus hőmérséklet, amely alatt a rendszer szinte teljesen rendezett (egyetlen nagy azonos spinekből álló klaszter dominál a rendszerben), fölötté pedig teljesen rendezetlen, nem alakulnak ki nagy klaszterek. Sok algoritmus létezik a rendszer tanulmányozására és a számunkra fontos átlagoknak a kiszámítására, a legtöbb viszont soros természetű. Pl. a Metropolis algoritmusban minden lépésben véletlenszerűen megváltoztatjuk egy spin állapotát. Ha az energiaváltozás ( $\Delta E$ ) negatív, akkor  $p=1$ , ha pozitív, akkor  $p=\exp(-\Delta E/kT)$  valószínűséggel fogadjuk el az új állapotot. Ha a CNN számítógépen akarjuk ezt az algoritmust implementálni, a cellák párhuzamos működése miatt kerülnünk kell, hogy közvetlen szomszédok állapotát ugyanabban a lépésben változtassuk meg. Ahhoz, hogy mégis kihasználhassuk a párhuzamoság előnyeit, egy sakktábla maszkot alkalmazunk, minden páros (páratlan) lépésben a maszk fehér (fekete) pontjainak megfelelő spinekre alkalmazzuk a Metropolis algoritmust. Ez a párhuzamosított algoritmus megegyezik azzal az esettel, mintha a Metropolis algoritmusban nem véletlenszerű hanem rendezett sorrendben választanánk a spineket. Ez az algoritmus így már könnyen implementálható a CNN számítógépen a véletlen-szám generátor és egyszerű logikai műveletek segítségével. A párhuzamosított algoritmust teszteltük digitális számítógépeken és az ACE16K CNN chipen. A kapott eredmények nagyon jól egyeznek a klasszikus Metropolis algoritmus eredményeivel. Ugyanakkor az algoritmus gyorsasága sok előnyt ígér a cellák számának a jövőbeli növelésével.

## 5. KÖVETKEZTETÉSEK

Amint a sok és különböző alkalmazási lehetőség is mutatja, az analogikai celluláris számítógépek egy teljesen új számítástechnikai paradigmát képviselnek. Speciális komplex műveleteket tudnak nagyon könnyen és gyorsan elvégezni, ezért egy jó kiegészítést adnak a manapság használt digitális számítógépeink mellé. Ezen számítógépek egy első lépést képviselnek a biológiai rendszerekből inspirált számítógép megalkotása felé.

## HIVATKOZÁSOK

- [1] G. Moore, Electronics, Vol. **38**, 114 (1965)
- [2] L. O. Chua, Lin Yang, *Cellular Neural Networks: Theory*, IEEE Transactions on Circuits and Systems, Vol. **35**, No. 10 (1988)
- [3] T. Roska, L. O.Chua, *The CNN Universal Machine: an analogic array computer*, IEEE Trans. CAS-II, vol. **40**, pp. 163-173, March (1993)
- [4] T. Roska, *Computational and computer complexity of analogic cellular wave computers*, Journal of Circuits, Systems and Computers, Vol. **12**, No.4. (2003)
- [5] T. Roska, L.O.Chua, Á.Zarándy, *Language, compiler, operating system for the CNN super-computer Berkeley*, Technical report UCB/ERL M93/34, (1993)

- [6] L. O. Chua and Tamas Roska, *Cellular neural networks and visual computing*, Cambridge University Press, (2002)
- [7] L. O. Chua, Lin Yang, *Cellular Neural Networks: Applications*, IEEE Transactions on Circuits and Systems, Vol. **35**, No. 10 (1988)
- [8] P. Kinget and M. Steyaert, *An analog parallel array processor for real-time sensor signal processing*, in Proc. of IEEE Int. Solid-state Circuits Conference, vol. **39**, pp. 92-93, (1996)
- [9] A. Zarandy, C. Rekeczky, *Bi-i: a standalone ultra high speed cellular vision system*, IEEE Circuits and Systems Magazine, Vol. **5**, p. 36-45, (2005)
- [10] K. R. Crouse, T. Yang, L. O. Chua, *Pseudo-random sequence generation using the CNN Universal Machine*, Fourth IEEE Int. Workshop on Cellular Neural Networks and their Applications, Seville, Spain, (1996)
- [11] T. Roska, L. O. Chua, D. Wolf, T. Kozek, R. Tetzlaff, F. Puffer, K. Lotz, *Simulating Nonlinear Waves and Partial Differential Equations via CNN – Part I: Basic Techniques, - Part II: Typical Examples*, IEEE Transactions on Circuits and Systems –I: Fundamental Theory and Applications, Vol. **42**, No. 10 (1995)
- [12] T. Kozek and T. Roska, *A double time-scale CNN for solving two-dimensional Navier-Stokes Equations*, International Journal of Circuit theory and Applications, Vol. **24**, 49-55 (1996)
- [13] M. Ercsey-Ravasz, T. Roska, Z. Neda, *Perspectives for Monte Carlo simulations on the CNN-UM*, Int. J. of Modern Physics C, Vol. **17**, No. 6, p. 909 (2006)
- [14] M. Ercsey-Ravasz, T. Roska, Z. Neda, *Stochastic Simulations on the Cellular Wave Computers*, Eur. Phys. J. B, Vol. **51**, No. 3, p. 407, (2006)