NEUROMORFIKUS SZÁMÍTÁSTECHNIKA, AVAGY HOGYAN VÁLTSUK APRÓPÉNZRE A 2024-ES FIZIKAI NOBEL-DÍJAT?

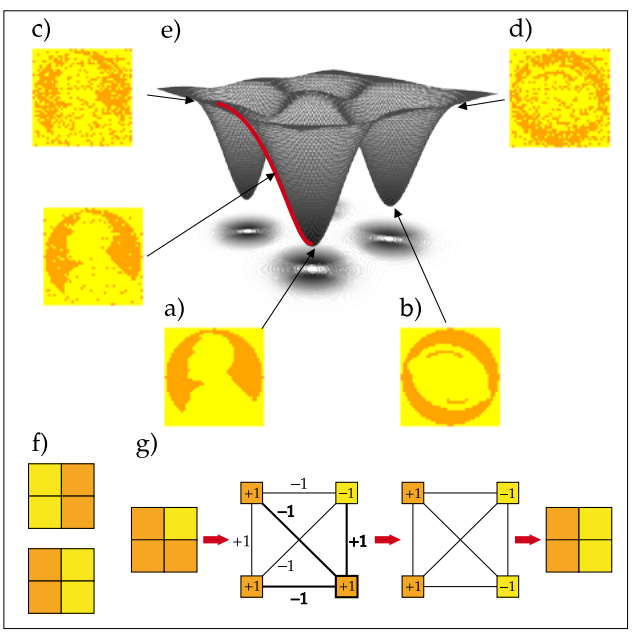
Fehérvári János Gergő, Balogh Zoltán, Halbritter András Ernő[@]
[1]Budapesti Műszaki és Gazdaságtudományi Egyetem, Fizika Tanszék, Budapest
[@]E-mail: halbritter.andras@ttk.bme.hu

Egy elemzés [7] szerint 2030-ra a világ teljes energiafogyasztásának több mint 20%-át a mesterséges intelligencián alapuló információs technológia fogja igényelni. Ahhoz, hogy az adatközpontok mégse emésszék fel a világ áramtermelését, a mesterséges intelligencia algoritmusainak fejlesztése mellett új, energiatakarékos hardvereszközökre is szükség van, melyek jobban követik a neurális hálózatok felépítését, és még az adattárolást és a számítást is egy helyen kezelik. Ezen kutatás-fejlesztési területbe, az ún. neuromorfikus számítástechnikába adunk betekintést a 2024-es fizikai Nobel-díj szemszögéből.

1. A Hopfield-féle neurális hálózat mint asszociatív memória

Játsszunk el a gondolattal, hogy egy elmosódott levelet kaptunk valamilyen díj elnyeréséről. Az érmet ábrázoló homályos képről *(1c., 1d. ábra)* kell eldöntenünk, hogy egy egetrengető felfedezésünkért Nobel-díjban részesültünk *(1a. ábra)*, esetleg valamelyik elfuserált projektünket citromdíjjal jutalmazták *(1b. ábra)*. Humán intelligenciánk nem hagy kétségek között őrlődni minket, első ránézésre felismerjük az elmosódott képeket. A 2024-ben fizikai Nobel-díjjal jutalmazott *John J. Hopfield*nek köszönhetjük azon mesterséges neurális hálózatok ötletét, amelyek hasonló képzettársításra, *asszociáció*ra képesek [1, 2].

A felismerésre szánt képeket persze először meg kell tanítanunk a hálózatnak. A tanítás során létrehozunk egy *térképet,* ami minden lehetséges, adott képpontokból álló képhez hozzárendel egyfajta *magasságot,* amit más néven a hálózat *energiájának* hívunk *(1e. ábra)*. Az energiatérképen látható gödrök legmélyebb pontjai az előre betáplált „tiszta” képeknek felelnek meg. A hálózat úgy működik, hogy minden egyes lépésnél kiválasztunk egy képpontot, és egy egyszerű szabállyal eldöntjük, hogy annak a színét a két lehetőség között (sárga vagy narancs) megváltoztassuk-e. A Hopfield által lefektetett szabállyal az energiatérképen minden egyes változtatásnál csak lefelé (alacsonyabb energia felé) tudunk mozogni (vagy változatlan energián maradni), így ha a Nobel-díjra „hasonlító” elmosódott képből indulunk ki, akkor a hálózat lépésről lépésre letisztítja a képet, és eljut a tiszta, eredetileg betáplált képhez *(1c., 1e., 1a. ábra)*. Ha a citromhoz hasonlító zajos képből indulnánk ki, akkor egy másik „gödörbe”, a letisztított citromképhez jutnánk *(1d., 1b. ábra)*.



1. ábra. A Nobel-díj és a citromdíj pixeles változatai, illetve a feltanított hálózat energiafelületének sematikus illusztrálása a különböző mértékben elmosódott képek függvényében. (f, g) Az asszociatív memória működésének szemléltetése két egyszerű képpel (f), és a bal alsó hibás pixel javításának lépéseivel (g)

Nézzük meg egy egyszerű példán az emlékalapú keresést matematikailag is! Vegyünk két 2 × 2 pixelből álló képet, ahol az egyiken a jobb, a másikon a bal oldalon található egy függőleges narancssárga vonal *(1f. ábra)* a citromsárga háttéren. Az ezen a két képen tanított hálózatot ekkor egy gráfként ábrázolhatjuk, melynek a csúcsai a képkockáknak feleltethetők meg *(1g. ábra)*. A hálózat működése során ezen csúcsokat hívjuk neuronoknak, melyek +1 vagy –1 állapotban lehetnek, a +1 felel meg a narancssárga, a –1 a citromsárga képkockának. A 4 neuront a gráfok élei, az úgynevezett mesterséges szinapszisok kötik össze, melyekhez tanulás során –1 és +1 közé eső valós számokat, úgynevezett *szinaptikus súly*okat rendelünk a lábjegyzetben ismertetett algoritmus alapján.[1] Ezek a súlyok kódolják az előre megtanított képeket. A példahálózatunkra a súlyok szintén az *1g. ábrá*ról olvashatók le.

Vegyünk egy elmosódott képet, ami a bal oldali narancssárga vonalat ábrázolná, de a jobb alsó sarokban egy pixelhiba miatt a citromsárga helyett narancssárga képpont található. Mutassuk meg ezt a képet a hálózatnak *(1g. ábra)*. Kiszemelve a hibás pixelt végezzünk el egy döntéshozatali lépést! Elsőként számoljuk ki az ebbe a neuronba érkező „neurális jelek” súlyozott összegét: vegyük az összes többi neuron (képpont) értékét, ezeket szorozzuk össze az adott képpontot a kiválasztott képponttal összekötő szinapszis (él) súlyával, és összegezzük fel a kapott értékeket, azaz az i-edik sorszámú neuronra az $a_i = \Sigma_{j \neq i} W_{i,j} \cdot x_j$ összeget számoljuk ki, ahol $W_{i,j}$ az i-edik és j-edik neuront összekötő súly, x_j a j-edik neuron értéke, a_i-t pedig az i-edik neuron aktivációjának nevezzük. Ha a képpontokat a bal felső saroktól az óramutató járásának megfelelően számozzuk, akkor a hibás képpont a harmadik pixel, ennek aktivációja:

$$a_3 = (-1) \cdot (+1) + (+1) \cdot (-1) + (-1) \cdot (+1) = -1 - 1 - 1 = -3.$$

Ha az aktiváció pozitív, a kiválasztott neuron értékét +1-re, ha negatív, akkor –1-re állítjuk. Rögtön látjuk, hogy a kiválasztott képpont ezen művelet hatására előjelet vált, és ezzel visszaáll a képpont helyes (citromsárga) értéke. Könnyen ellenőrizhetjük, hogy az *1f. ábrán* bármelyik kép bármelyik képpontját kiválasztva ezek előjele nem változik, vagyis a két előre betáplált képet valóban két stabil megoldásnak (energiaminimumnak)

[1] Egy képünk összes képpontját számozzuk be (azaz az $n \times n$ képpontú képmátrixból készítsünk egy $N = n \cdot n$ hosszúságú képvektort). Ezután az i indexű képponthoz (azaz az i-edik neuronhoz) a színének megfelelően rendeljünk hozzá $x_i = \pm 1$ értéket. Egy adott képből az i-edik és j-edik neuron közötti $W_{i,j}^{\text{kép}}$ súlyt az $x_i \cdot x_j$ szorzat adja. Több kép tanítása esetén az egyes képeknek megfelelő $W_{i,j}^{\text{kép}}$ súlyokat átlagoljuk a különböző képekre, így kapjuk meg a neurális hálózat $W_{i,j}$ szinaptikus súlyait. Fontos, hogy a neuronok önmagukkal nincsenek összekötve, azaz $W_{i,i} = 0$. Az adott neuronállapothoz és az előre beprogramozott súlyokhoz rendelt energiát $E = -(1/2)\Sigma_{i,j} x_i \cdot W_{i,j} \cdot x_j$ képlet szerint számoljuk.

tekinthetjük, míg bármelyik pixel hibás megjelenítése esetén a hálózat az adott pixelt javítja.

A súlyok szemléletesen azt mondják meg, hogy a hálózati működés során két neuron ad-e egymásnak információt, és ha igen, milyet. Nulla súly esetén a kiszemelt két neuron nem beszélget egymással. Nem nulla súly esetén beszélnek, sőt ha pozitív ez a súly, akkor ugyanolyan állapotba, ha negatív akkor pedig ellentétes állapotba való beállásra ösztönzik egymást.

A 2×2 képpontos példán bemutatott hálózati működés szabályai szerint járunk el sokkal nagyobb méretű képeknél is, csak ekkor a képpontoknak megfelelő neuronokat egymás után frissítgetve, akár az összes neuron többszöri frissítése során sokkal több lépésben jutunk el az energiaminimumot jelentő megoldáshoz.

2. A Hopfield-féle neurális hálózatok általánosítása valószínűségi optimalizálási feladatokra

Az előző példában a Hopfield-hálózat a legközelebbi minimumhely megtalálásával az előre betáplált képekre „asszociált” *(1a–e. ábra)*. A Hopfield-hálózat megalkotása után nem sokkal kiderült, hogy bizonyos egzaktul nem megoldható matematikai feladatokat meg lehet úgy fogalmazni, mint egy ilyen energiafelület legmélyebb pontjának megtalálása. Vagyis ha a Hopfield-neuronok közötti kapcsolatba képek helyett egy matematikai optimalizációs feladatot kódolunk, akkor ugyanezzel az iterációs módszerrel a hálózat képes a megoldást megtalálni. Vegyük azonban észre, hogy míg a képek tárolásánál minden minimumhely egy külön emlék lehetett, és örültünk, hogy ezek szétváltak (különben a hálózat nem tudná az emlékeket megkülönböztetni), addig az optimalizációs feladatnál a legjobb megoldást, vagyis legeslegmélyebb pontot keressük. Felvetődik tehát az igény, hogy úgy befolyásoljuk a hálózati működést, hogy az ne a legközelebbi szélsőértéken vagy energiagödörben álljon meg (*2a. ábra,* szürke pont), hanem eljusson a globálisan optimális megoldáshoz, a legmélyebb gödörhöz (*2a. ábra,* piros pont).

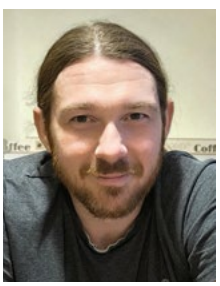
Ezt a működés randomizálásával, vagyis véletlen faktorok becsatolásával érhetjük el. Egyszerűen, amikor meghozunk egy döntést, akkor nem az a_i aktiváció előjelét nézzük meg, hanem az a_i aktivációhoz hozzáadunk egy véletlen számot, és az így kapott összeg előjelének megfelelően frissítjük a neuron értékét. Ennek köszönhetően a hálózat az energiát nem szigorúan, minden egyes lépésben csökkenti, hanem időről időre energianövelő lépéseket is végez. Ez pedig lehetővé teszi, hogy ne a legközelebbi minimumba érkezzen meg, hanem több szélsőértéket is felfedezzen, ezzel növelve esélyeit az energiatérkép legmélyebb pontjának megtalálásának. Ezzel a módszerrel természetesen nem lehet a megoldást biztosan megtalálni, de a kisorsolt véletlen számok tartományának megfelelő megválasztásával elérhető, hogy nagy valószínűséggel megtaláljuk a megoldást.

A véletlen aktiváció gondolatát a 2024-es fizikai Nobel-díj másik díjazottja, *Geoffrey Hinton* vitte tovább, aki a *Boltzmann-gép* megszerkesztésével olyan neurális hálózatot alkotott, amely már felépítéséből fakadóan is használ véletlen faktorokat, és ennek köszönhetően még szélesebb körben alkalmazható [4].

A fenti módszerrel egy bonyolultnak számító optimalizációs feladat, a gráfok maximális vágásának a megkeresése (max cut) is megoldható [5]. Itt egy gráf csúcspontjait *(2b. ábra)* kell úgy két csoportba osztani (narancssárga és citromsárga, lásd *2c. ábra*), hogy a két csoportot összekötő élek száma maximális legyen. Megmutatható, hogy számos gyakorlati optimalizációs feladat visszavezethető a maximális vágási probléma megoldására, például az integrált áramkörök tervezése, logisztikai feladatok megoldása vagy gyártási optimalizáció területén [6].

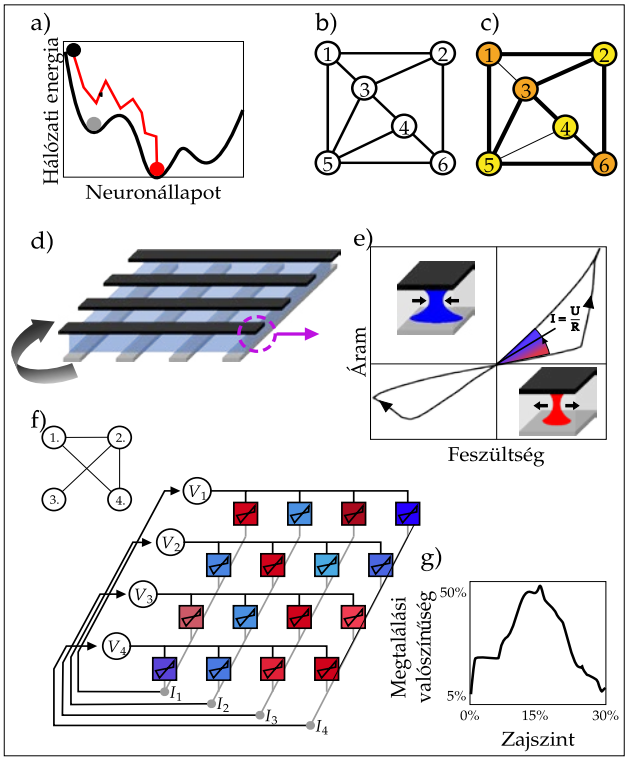
A *2c. ábrán* a *2b. ábrán* szereplő gráf maximális vágásának megoldását ábrázoljuk, úgy, hogy a narancssárga és citromsárga csúcspontok között futó élek vastagon vannak jelölve, így csak a vastag élek számát kell leszámolni, ami jelen esetben 8. Ez a feladat tetszőlegesen nagy gráfra megoldható próbálgatással, hiszen egy adott állapotnál az ellentétes színű csúcsokat összekötő élek száma könnyen megszámolható. Azonban N csúcs esetén az összes lehetséges variáció végigpróbálása 2^N

Fehérvári János Gergő a BME Fizika Tanszék Kooperatív Doktori Program (KDP) ösztöndíjas doktorandusza a Semilab Zrt.-vel együttműködésben. Kutatási témája atomerő-mikroszkópiai módszerek fejlesztése félvezetőipari alkalmazásokhoz, melynek része a memrisztorok fizikájának vizsgálata vezető atomerő-mikroszkópiával. OTDK-nyertes TDK-dolgozatában memrisztív Hopfield-féle neurális hálózatokat vizsgált.



Balogh Zoltán egyetemi docens (BME Fizika Tanszék). Kutatási területei: memrisztorok, valamint atomi és molekuláris rendszerek zajjelenségeinek vizsgálata. Korábbi Bolyai János Kutatói Ösztöndíjas, jelenleg a „Nanoelektronikai eszközök elektromos zajának feltárása, testreszabása és hasznosítása” című OTKA FK-pályázat vezetője.

Halbritter András Ernő a BME Fizika Tanszék egyetemi tanára, az MTA doktora. 2012 és 2024 között a Fizika Tanszék tanszékvezetője, jelenleg a BME Fizikai Intézet igazgatóhelyettese. A HUN-REN–BME Kondenzált Anyagok Fizikája kutatócsoport vezetője. Kísérleti kutatócsoportja neuromorfikus elektronikával, illetve atomi és molekuláris elektronikával foglalkozik.



2. ábra. A maximális vágási probléma megoldása memristztív Hopfield-féle neurális hálózatok segítségével. (a) A Hopfield-hálózat energiatérképében a globálisan legjobb megoldást (piros pont) csak úgy találhatjuk meg, ha az aktiváció számításánál véletlenszerűséget, vagy zajt is bevezetünk. (b) Hatcsúcsú gráf, melynek a maximális vágását keressük. (c) A maximális vágási probléma megoldása. A Hopfield-hálózatban a neuronok narancssárga (+1) és citromsárga (–1) értéke adja meg, hogy melyik csoportba soroljuk a csúcspontot, míg a $W_{i,j}$ súlyoknak egy, illetve zérus értéket adunk, ha az i-edik és j-edik csúcs össze van kötve vagy nincs összekötve [3]. (d, e, f) Memristív neuromorfikus hardverek. Memristorok $I(V)$ görbéje (e): kis feszültségeken az áram-feszültség kapcsolat lineáris, a megfelelő küszöbfeszültségnél viszont egy kapcsolási folyamat indul el: pozitív feszültségnél a két elektróda közötti filamentum elkezd nőni, majd megfelelő negatív előfeszítésnél a folyamat megfordítható, a filamentum átmérője csökken. (d) Kereszthuzalozású memóriahálózat készítése memristorokból. (f) A panel bal felső sarkában látható gráfnak megfelelő Hopfield-féle neurális hálózat megvalósítása kereszthuzalozású memristorhálózattal. (g) A maximális vágási probléma megoldásának megtalálási valószínűsége 60×60-as memristív Hopfield-féle neurális hálózatban a zajszint függvényében [3]

műveletet igényel, ami 1000 csúcspont esetén $3{,}4 \cdot 10^{284}$ év számolási időt jelent, feltételezve hogy egyetlen műveletet 1 ns = 10^{-9} s alatt el tudunk végezni. A fenti, Hopfield-hálózattal végzett valószínűségi optimalizálás ennél jóval hatékonyabb megoldást jelent, azonban ennél a megközelítésnél is $\approx N^2$ szorzást és összeadást el kell végezni ahhoz, hogy egy iterációs körben az összes neuron értékét egyszer frissítsük. Ez ezer neuron esetén egymillió szorzást jelent. A későbbiekben olyan, ún. neuromorfikus hardvereszközöket mutatunk be, melyekkel egymillió szorzás helyett egyetlen lépésben kiértékelhető az összes neuron aktivációja, így a hagyományos számítógépeken megvalósított Hopfield-hálózatok helyett lényegesen gyorsabban és energiahatékonyabban végzik el az optimalizálást.

3. A memrisztorhálózatok mint a mesterséges intelligencia hatékony hardvergyorsítói

Általánosan is elmondható, hogy a mesterséges intelligencia alapját képező mesterséges neurális hálózatok nagyon bonyolult feladatok megoldására képesek, de futtatásuk hagyományos digitális számítógépeken elképesztő számítási igénnyel jár, hiszen nagyon-nagyon sok neurális kimeneti értéket és szinaptikus súlyt kell összeszorozni és összeadni. Ezenkívül az adattárolás és a műveletvégzés fizikailag szeparált egységekben történik, így a memória és a processzor közötti adatmozgatás is szűk keresztmetszetté válik. A ChatGPT-nek a maga 175 millió paraméterével és a Hopfield-hálózatnál jóval összetettebb felépítésével saját bevallása szerint 10–20 trillió (milliárdszor milliárd) számítási műveletet kell elvégeznie egyetlen kérdésünk megválaszolásához. Ez a számítási igény már energiafogyasztásban is számottevő, egy elemzés szerint 2030-ra a világ teljes energiafogyasztásának több mint 20%-át a mesterséges intelligencián alapuló információtechnológia fogja igényelni [7]. Emiatt a mesterséges intelligencia további fejlődéséhez nem elég az algoritmusokat fejlesztünk és bonyolultabb feladatokat megoldani képes hálózatokat kigondolnunk. Legalább ugyanilyen fontos az, hogy a háttérben meghúzódó számításokat is hatékonyan, energiatakarékosan tudjuk elvégezni újfajta hardvereszközökkel, melyek felépítésükben jobban követik a neurális hálózatok struktúráját, és akár egy helyen meg tudják oldani az adattárolást és a számítást.

Ezen új, ún. *neuromorfikus* hardverek fejlesztéséhez a biológiai idegrendszer szolgáltat inspirációt, hiszen gondoljuk csak bele, hogy az emberi agy mintegy 30 W-os energiafogyasztás mellett milyen hihetetlen hatékonyságra képes!

3.1. A memrisztorok mint mesterséges szinapszisok

Az ilyen neuromorfikus hardverek lehetséges építőkövei az úgynevezett *memristor*ok, azaz memóriával rendelkező ellenállások. Ezek olyan fizikai rendszerek, melyek ellenállása nagy feszültséggel hangolható, kis feszültséggel pedig információvesztés nélkül olvasható. Fizikailag egy memristort úgy valósítunk meg, hogy két fém elektróda közé egy speciális, eredetileg szigetelő réteget helyezünk. Megfelelően nagy elektromos feszültséggel a szigetelőben ionokat, pl. ezüstionokat tudunk mozgatni úgy, hogy adott polaritású feszültséggel az ezüstionokból egy vékony vezetéket (filamentumot) építünk fel a két elektróda között. Ellentétes polaritású feszültséggel viszont leépítjük a filamentumot, az egyre kisebb átmérőjűvé válik, majd végül akár meg is szűnik. Ezt a reprodukálhatóan elvégezhető folyamatot szemlélteti a *2e. ábrán* bemutatott $I(V)$ áram-feszültség karakterisztika. Itt a kapcsolás egy kisebb ellenállású (kék) és egy nagyobb ellenállású (piros) állapot között történik, de a megfelelő feszültség alkalmazásával gyakorlatilag folytonosan han-

golhatjuk a filamentum átmérőjét, és az ennek megfelelő R ellenállást (lásd a piros-kék átmenetben szemléltetett tetszőleges meredekségű $I(V)$ görbét a *2e. ábrán*). Alacsony feszültségen viszont a rendszer megőrzi az állapotát, azaz az ellenállásállapot kiolvasható az ellenállás megváltoztatása nélkül. Vegyük észre, hogy a memristorok analóg memóriának felelnek meg, amik kiválóan alkalmasak arra, hogy a szinte folytonosan hangolható $G = 1/R$ vezetőképességükben adatot tároljunk. A következőkben pedig megmutatjuk, hogy akár mesterséges neurális hálózatok szinaptikus súlyfaktorainak reprezentálására is lehet használni őket.

3.2. Kereszthuzalozású memristorhálózatok és memristív Hopfield-hálózatok

A memristorok információtechnológiai felhasználásához azonban hálózatba is kell rendezni őket. Ez meglepően egyszerűen megvalósítható, elég a memristorok kialakítására alkalmas szigetelőréteget a *2d. ábrán* látható szürke és fekete fém elektródák közé helyezni. Egy adott szürke és fekete elektródapárra feszültséget kapcsolva a két elektróda találkozásánál kialakul a memristorkontaktus, aminek a vezetőképessége szinte tetszőlegesen beállítható.

Memristív eszközök és hálózatok nagyon sokféle anyagcsaládból, különböző architektúrákban hozhatók létre [8–10]. A továbbiakban a Hopfield-féle neurális hálózatok példáján vizsgáljuk meg, hogy a *2d. ábrán* látható kereszthuzalozású memristorhálózatok hogyan is alkalmazhatóak hatékony hardvergyorsítóként. Először is a hálózat súlyait kell beállítani a megoldandó problémának megfelelően. A *2f. ábra* memristív hálózata az ábra bal felső sarkában látható egyszerű, négycsúcsú gráfot valósítja meg, melyen az 1–2, 1–4, 2–3 és 2–4 csúcsok vannak összekötve. Ezt úgy programozhatjuk be a keresztvonalas hálózatba, hogy a felső fekete és az alsó szürke elektródákat beszámozzuk, és azon számú elektródák között alakítunk ki nagy vezetőképességű memristív kontaktust (*2f. ábra,* kék memristorok), ahol a gráf megfelelő számú csúcsai is össze vannak kötve, míg a többi fekete és szürke elektróda közé kifejezetten kis vezetőképességű memristorkontaktust alakítunk ki (*2f. ábra,* piros memristorok). Az adott neuron állapotát pedig a fekete elektródákra kapcsolt V_1, V_2, V_3, V_4 feszültségek jellemzik: +1 állapotban $+V$, míg –1 állapotban $-V$ feszültséget alkalmazunk, azaz az i-edik elektródára kapcsolt V_i feszültség arányos az i-edik neuron x_i állapotával. A szürke vezetékeken pedig az adott neuron aktivációját határozhatjuk meg az ott mérhető áramon keresztül, ami alapján eldöntjük, hogy az adott neuron állapotát megváltoztassuk-e. Válasszunk ki egy adott neuront a *2f. ábrán,* például az elsőt! Mivel a különböző elektródákból érkező áramkomponensek összeadódnak, az első szürke vezetékre

$$I_1 = G_{1,1} \cdot V_1 + G_{1,2} \cdot V_2 + G_{1,3} \cdot V_3 + G_{1,4} \cdot V_4$$

áram adódik. Itt a $G_{1,2}$, $G_{1,3}$ és $G_{1,4}$ vezetőképességek az 1–2, 1–3 és 1–4 neuronok közötti $W_{1,2}$, $W_{1,3}$, $W_{1,4}$ szinaptikus súlyokat reprezentálják, amik a *2f. ábra* bal felső gráfján szereplő összeköttetések alapján rendre nagy, kicsi és nagy vezetőképességnek felelnek meg. A Hopfield-hálózatban a neuronok önmagukkal nincsenek összekötve, így $G_{1,1}$ egy nagyon kicsi, szinte nulla vezetőképességnek felel meg. Jól látszik, hogy a szürke vezetékeken mért áramok valóban az adott neuron aktivációját, azaz a többi neuron állapotának az összekötő súlyokkal vett súlyozott összegét adják meg.

Rögtön látható, hogy egy N neuronból álló memristív hálózatban az összes neuron aktivációja egyszerre, egyetlen lépésben *kiszámolódik* az Ohm-törvény és a Kirchhoff-törvények alapján, azaz a digitális számítógépekkel ellentétben nincs szükség $\approx N^2$ szorzási művelet szoftveres elvégzésére és az ehhez szükséges memória és processzor közötti adatmozgatásra. A neurális állapotok ún. vektorának és az összekötéseket jelentő szinapszisok ún. mátrixának összeszorzása egy alapművelet mindenfajta neurális hálózatban, így a szoftveresen N^2 műveletet igénylő vektor-mátrix szorzás leegyszerűsítése egylépéses hardveralapú vektor-mátrix szorzássá jelentős energiamegtakarítást jelent tetszőleges neurális hálózatban. Ez manapság már nemcsak elvi alapon működik, hanem többször tízmillió memristor hálózatba építésével olyan mesterséges neurális hálózatokat lehet építeni, melyek meghökkentő energiahatékonyság mellett képesek komplex képfelismerési vagy egyéb számítási feladatok nagy pontosságú elvégzésére [9, 11], és a fenti példában szereplő maximális vágási probléma is hatékonyan megoldható nagyméretű memristív Hopfield-hálózatok segítségével [12].

3.3. A memristorok zaja

Visszatérve a maximális vágási probléma Hopfield-féle neurális hálózatos megoldásához, felmerül a kérdés, hogy a valószínűségi optimalizáláshoz szükséges véletlenszerűséget *(2a. ábra)* egy hardveres memristív hálózatban *(2f. ábra)* hogyan tudjuk biztosítani. Szerencsére a memristorok nemcsak mesterséges szinapszisnak tekinthetők, hanem hangolható zajforrásként is működnek, azaz a memristotorok vezetőképesség-állapotának beállítása közben az adott állapot vezetőképességének időbeli fluktuációja is változik, az utóbbi mennyiséget a G vezetőképesség ΔG időbeli szórásával írhatjuk le. Tipikusan minél kisebb G vezetőképességet állítunk be, annál nagyobb lesz a $\Delta G/G$ relatív vezetőképesség-zaj. Ez azt jelenti, hogy a *2f. ábrán* a kék, jól vezető memristorok vezetőképességét úgy választhatjuk meg, hogy a vezetőképesség relatív zaja optimális legyen a maximális vágási probléma megoldásának megtalálásához. Egy korábbi tanulmányunkban ezt a kérdést vizsgáltuk, azaz hogy különböző, valós memristív rendszereken mért zajszintek mellett a memristorhálózat megadott számú lépés elvégzése után milyen arányban, milyen va-

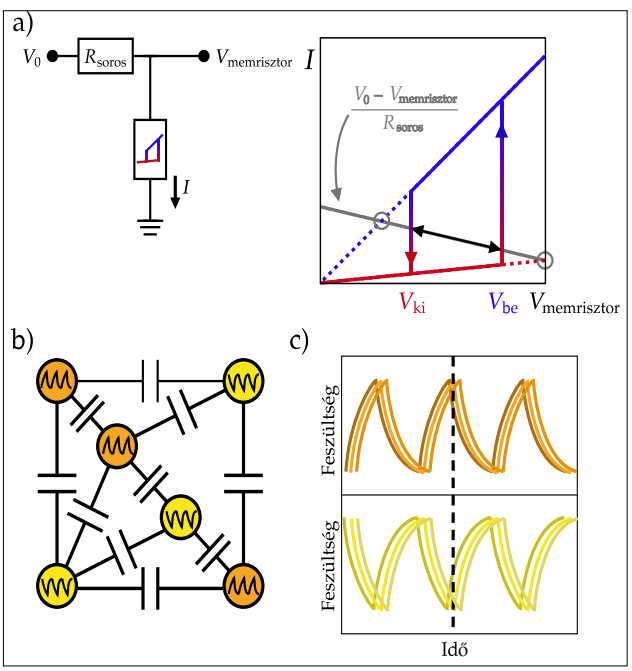
lószínűséggel képes az optimális megoldást megtalálni. Tapasztalataink egyik konklúziója a *2g. ábrán* látható, megmutatva, hogy az optimális működést meglepően magas zajszintnél érjük el, amikor a memrisztoregységek vezetőképességének időbeli fluktuációja az átlagos vezetőképesség 15%-át is eléri [3].

Míg a valószínűségi optimalizáláson alapuló neurális hálózatokban a működés lényegi részét adja a memristorok vezetőképességének jelentős időbeli fluktuációja, a hagyományos, determinisztikus működést nyújtó neurális hálózatokban a memrisztoregységek zaja jelentős problémát jelent, hiszen a szinaptikus súlyok felbontásának nyilvánvaló korlátja a memristorok vezetőképességének időbeli ingadozása. A memristorokban jelentkező fluktuációk kísérleti vizsgálata, megértése és csökkentése kutatócsoportunkban kiemelt kutatási terület [13–15]; illetve a szakirodalomban is kulcsfontosságú volt a zajcsökkentés a világrekordnak számító 11 bites súlyfelbontás elérésében [16].

4. És ha az egész számítást a fizika végezné el?

A fenti példában a memristorok mesterséges szinapszisként működtek, és alapvetően a neurális értékek és szinaptikus súlyok egylépéses, hardveralapú összeszorzásában nyújtottak kifejezetten energiahatékony alternatívát a hagyományos digitális számításokhoz képest. Ettől függetlenül, a bemutatott memristív Hopfield-féle neurális hálózatokban az aktivációk kiolvasását és a neurális állapotok ennek megfelelő frissítését továbbra is hagyományos számítógépek végezték [12]. Felmerül a kérdés, hogy elképzelhetőek-e olyan számítási módszerek, ahol hagyományos számítógépek helyett egy teljesen önműködő fizikai rendszer végzi el a számítást. Memristorok segítségével ilyen eszközök is építhetők. Ebben az esetben ún. illékony memristorokat kell használni, pl. a kutatócsoportunkan is vizsgált Mott-fázisátalakuláson alapuló VO_2 memristorokat [17–19]. Míg a *2e. ábrán* bemutatott nem illékony filamentáris memristorokra mint a biológiai szinapszisok mesterséges megfelelőire gondolunk, addig a VO_2 memristorok a biológiai neuronokhoz hasonló működést mutatnak, velük számos neurális jelalak létrehozható [20]. Az egyik legegyszerűbb alkalmazásuk ún. oszcillátoráramkörök készítése, melyekben egy egyenfeszültség hatására a VO_2 alapú áramkör a ki- és bekapcsolt állapotok közötti önműködő váltogatással periodikus feszültségoszcillációt mutat. Ezt a folyamatot a *3a. ábrán* érthetjük meg, ahol a jobb oldali panel az illékony VO_2 memristor $I(V)$ görbéjét szemlélteti. Zérus feszültségből indulva egy nagy ellenállású állapotban van a rendszer (piros egyenes), majd a V_{be} bekapcsolási feszültségnél egy lokális szigetelő-fém átalakulás zajlik le az aktív tartományban, és a memristor átkapcsol egy kisebb ellenállású állapotba (kék egyenes). A feszültséget visszacsökkentve némi hiszterézissel, V_{ki} feszültségnél visszakapcsol a rendszer az eredeti nagy ellenállású állapotba. Az ilyen oszcillátor-áramkörökben egy nagy soros ellenállást is alkalmazunk, amin keresztül egy V_0 egyenfeszültséget kapcsolunk a rendszerre (*3a. ábra,* bal oldal). Ekkor az áramkörben $I = (V_0 - V_{\text{memristor}})/R_{\text{soros}}$ áram alakul ki, ahogy a jobb oldali szürke vonal mutatja. A két (piros és kék) ellenállás-állapotnak megfelelő lehetséges áramértékek (szürke körök) viszont instabil állapotnak felelnek meg: mire elérnénk a jobb oldali szürke kört, az áramkör már bekapcsol, ekkor viszont hirtelen lecsökken a feszültség a soros ellenállással történő feszültségosztás miatt, de még a bal oldali szürke kör elérése előtt visszakapcsol a rendszer a nagy ellenállású állapotba, és kezdődik az egész elölről. Így a meghajtó egyenfeszültség a memristor feszültségének periodikus oszcillációját eredményezi, a periódusidőt pedig egy memristorral párhuzamosan kötött kapacitással tudjuk beállítani.

2024-ben az IBM Zürich kutatói demonstráltak olyan áramköröket, melyek a maximális vágási problémát csatolt VO_2 oszcillátorokkal tudják önműködően megoldani [21]. Ebben a megközelítésben a neuronok a VO_2 oszcillátoráramkörök, melyeket a vizsgált gráfon szereplő összeköttetések szerint kondenzátorokkal összecsatolunk, azaz a *3b. ábra* szerint valósítjuk meg a *2b. ábrán* levő gráfot. A csatolás hatására az oszcillátoráramkörök szinkronizálódnak, és egy részük az első oszcillátorral azonos fázisban, míg másik részük az első oszcillátorral ellentétes fázisban kezd el oszcillálni (lásd a *3c. ábrán* a



3. ábra. A maximális vágási probléma megoldása oszcillátorhálózatokkal. (a) A fém-szigetelő átalakuláson alapuló vanádium-oxid illékony memristorok működési sémája. (b, c) A maximális vágási probléma megoldását kapacitívan csatolt oszcillátorok segítségével kaphatjuk meg az azonos, illetve ellenfázisba beálló oszcillátorok szerint választjuk szét a vizsgált gráf csúcsait. A fázisok két lehetséges beállását segíti, ha a közös oszcillációs frekvencia kétszeresének megfelelő szinuszos jelet is csatolunk az oszcillátorhálózathoz [21]

fázisban levő narancs és az ellenfázisban levő citromsárga oszcillációkat). A maximális vágási feladat megoldását a fázisok rendeződése adja. Az azonos fázisban működő oszcillátorokat színezzük narancssárgára, míg az ellentétes fázisban működő oszcillátorokat citromsárgára, ami meg is adja a *2c. ábrán* már bemutatott megoldást. Ezzel a módszerrel egyelőre maximum kilenc oszcillátort tartalmazó csatolt áramkört sikerült megvalósítani mintegy 2 kHz oszcillációs frekvencia mellett, helyes megoldást adva a maximális vágásra [21]. Ezen a területen kutatócsoportunk a működés felgyorsításának kutatásán dolgozik. Az egyedi VO_2 memrisztív eszközöket az ETH Zürich laboratóriumával együttműködve sikerült 15 ps idő alatt bekapcsolni, ami egészen meghökkentő, pár fJ-os kapcsolási energiának felel meg [19]. Ezen extrém kapcsolási idők oszcillátoráramkörökben bizonyos fizikai korlátok miatt nem érhetők el, de kutatásaink alapján az eddigi VO_2 oszcillátoroknál lényegesen gyorsabb, 100 MHz-es oszcillációs frekvenciák már megvalósíthatók [22, 23].

5. Összegzés

A 2024-es fizikai Nobel-díjhoz kapcsolódóan egy speciális optimalizálási feladat, a maximum vágási probléma mentén tekintettünk át olyan lehetséges megoldási sémákat, melyek jól szemléltetik az információtechnológia fejlődési irányait. A próbálgatásos megoldás könnyen beprogramozható egy hagyományos számítógépbe, azonban a gráf méretével exponenciálisan elszáll a megoldási idő. A Hopfield-féle neurális hálózatok egy hatékony valószínűségi optimalizálási megoldást nyújtanak, de szoftveres megoldás esetén minden egyes iterációs körben N^2 szorzási műveletet kell elvégezni, ami nagy gráfméretnél erőforrásigényes feladat. Ehhez képest egy lényegesen energiahatékonyabb megoldást kapunk, ha a neurális értékek és súlyok szorzását hardveresen, egyetlen lépésben végezzük el kereszthuzalozású memrisztorhálózatokkal. Végezetül egy elegáns megközelítést is bemutattunk, ahol a problémát a fizika oldja meg csatolt oszcillátorok szinkronizálódásán keresztül.

Nyilvánvaló, hogy korunk exponenciálisan növekvő számítási igényeit a Neumann-féle számítógépek nem fogják tudni kezelni; a hagyományos számítástechnikát ki kell egészíteni kifejezetten energiahatékony célhardverekkel illetve hardvergyorsítókkal, melyek akár peremeszközként, az adatok keletkezési helyén képesek „aprópénzből üzemeltetett", minimalizált fogyasztású adatelemzésre. A terület hihetetlen fejlődését látva számíthatunk rá, hogy az újfajta, *neuromorfikus* hardverek a közeljövőben megjelennek mindennapi eszközeinkben. Látva pedig, hogy ebben a technológiai fejlődésben az oszcillátoralapú neurális hálózatok is fontos szereplővé válhatnak, ne feledkezzünk meg arról, hogy a Neumann-féle számítógépek korlátait esetlegesen feloldó oszcillátorhálózatok ötletéhez maga Neumann János is alapvetően járult hozzá [24, 25].

Irodalom

1. Hopfield J. J. (1982): Neural networks and physical systems with emergent collective computational abilities. Proceedings of the National Academy of Sciences, 79(8), 2554–2558.
2. Hopfield J. J., Tank D. W. (1986): Computing with neural circuits: A model. Science, 233(4764), 625–633.
3. Fehérvári J. G., Balogh Z., Török T. N., Halbritter A. (2024): Noise tailoring, noise annealing, and external perturbation injection strategies in memristive Hopfield neural networks. APL Machine Learning, 2(1), 016107, 01.
4. Hinton G. E., Sejnowski T. J., Ackley D. H. (1984): Boltzmann machines: Constraint satisfaction networks that learn.
5. Zhang J. L., Wu L. Y., Zhang X. S. (2001): Application of discrete Hopfield-type neural network for max-cut problem. In: ICONIP, pp. 1439–1444.
6. Nieberg F. T., Pardella G. (2011): Minimization in VLSI Chip Design Application of a Planar Max-Cut Algorithm.
7. Jones N. (2018): How to stop data centres from gobbling up the world's electricity. Nature, 561, 163–166.
8. Ielmini D., Waser R. (2016): Resistive Switching. John Wiley & Sons, Ltd.
9. Aguirre F., et al. (2024): Hardware implementation of memristor-based artificial neural networks. Nature Communications, 15(1), 1974.
10. Xia Q., Yang J. J. (2019): Memristive crossbar arrays for brain-inspired computing. Nature Materials, 18, 309–323.
11. Huang Y., et al. (2024): Memristor-based hardware accelerators for artificial intelligence. Nature Reviews Electrical Engineering, 1(5), 286–299.
12. Cai F. (2020): Power-efficient combinatorial optimization using intrinsic noise in memristor Hopfield neural networks. Nature Electronics, 3, 409–418, 07.
13. Balogh Z., Mezei G., Pósa L.,Sánta B., Magyarkuti A., Halbritter A. (2021): 1/f noise spectroscopy and noise tailoring of nanoelectronic devices. Nano Futures, 5(4), 042002.
14. Sánta B., Balogh Z., Pósa L., Krisztián D., Török T. N., Molnár D., Sinkó Cs., Hauert R., Csontos M., Halbritter A. (2021): Noise tailoring in memristive filaments. ACS Applied Materials & Interfaces, 13(6), 7453–7460,. PMID: 33533590.
15. Sánta B., Balogh Z., Gubicza A., Pósa L., Krisztián D., Mihály Gy., Csontos M., Halbritter A. (2019): Universal 1/f type current noise of Ag filaments in redox-based memristive nanojunctions. Nanoscale, 11, 4719–4725.
16. Rao M., et al. (2023): Thousands of conductance levels in memristors integrated on CMOS. Nature, 615(7954), 823–829.
17. Pósa L., Hornung P., Török T. N., Schmid S. W., Arjmandabasi S., Molnár Gy., Baji Zs., Dražić G., Halbritter A., Volk J. (2023): Interplay of thermal and electronic effects in the Mott transition of nanosized VO_2 phase change memory devices. ACS Applied Nano Materials, 6(11), 9137– 9147.
18. Molnár D., Török T. N., Kövecs R., Pósa L., Balázs P., Molnár Gy., Jimenez Olalla N., Leuthold J., Volk J., Csontos M., Halbritter A. (2023): Autonomous neural information processing by a dynamical memristor circuit. https://arxiv.org/abs/2307.13320
19. Schmid S. W., Pósa L., Török T. N., Sánta B., Pollner Zs., Molnár Gy., Horst Y., Volk J., Leuthold J., Halbritter A., Csontos M. (2024): Picosecond femtojoule resistive switching in nanoscale VO_2 memristors. ACS Nano, 18(33), 21966–21974.
20. Yi W., et al. (2018): Biological plausibility and stochasticity in scalable VO_2 active memristor neurons. Nature Communications, 9(1), 4661.
21. Maher O., et al. (2024:) A CMOS-compatible oscillation-based VO_2 Ising machine solver. Nature Communications, 15(1), 3334.
22. Pollner Zs. S. (2023): Ultragyors, memrisztív oszcillátor-áramkörök fejlesztése. TDK dolgozat, Budapesti Műszaki és Gazdaságtudományi Egyetem.
23. Pollner Zs. S. (2024): VO_2 oszcillátor nagy sebességű oszcillációs neurális hálózathoz. TDK dolgozat, Budapesti Műszaki és Gazdaságtudományi Egyetem.
24. von Neumann J. (1954): Non-linear capacitance or inductance switching, amplifying and memory devices.
25. Wigington R. L. (1959): A new concept in computing. Proceedings of the IRE, 47(4), 516–523.