ChatGPT

“Spiking Neural Networks” Makalesinin Derinlemesine Analizi

Makalenin Konusu ve Temel Amacı

Bu makalenin konusu, **Spiking Sinir Ağları (SNN)** ile geleneksel **Yapay Sinir Ağlarının (ANN)** enerji verimliliğinin adil ve kapsamlı bir şekilde karşılaştırılmasıdır. Özellikle, SNN’lerin olay güdümlü (event-driven) ve ikili “ateşleme” (spike) sinyalleriyle çalışması sayesinde daha az hesaplama yüküyle yüksek enerji verimliliği vadettiği iddiası sorgulanmaktadır [1] [2]. Makalenin temel amacı, SNN’lerin gerçekten **daha enerji-verimli olup olmadığını** ve eğer öyleyse **hangi koşullar altında** olduğunu objektif bir analizle ortaya koymaktır. Bu doğrultuda çalışma, geçmişteki enerji değerlendirmelerinde yapılan basitleştirici varsayımları (ör. sadece toplama/çarpma saymak, bellek erişimlerini yok saymak) yeniden ele alarak SNN’lerin enerji avantajlarını **bütüncül bir donanım perspektifinden** değerlendirmektedir [2] [3].

Makale, **SNN ile eşdeğer bir kantize ANN (QNN)** modeli tanımlayarak ikisi arasında adil bir karşılaştırma zemini kurmayı hedefler. Yazarlar, T adım uzunluğundaki bir SNN’nin, aktivasyonları $\lceil \log_2(T+1)\rceil$ bit ile temsil edilen bir **kantize yapay sinir ağı** (QNN) ile aynı bilgi temsil gücüne sahip olduğunu gösteriyor [4] [5]. Bu sayede her SNN için benzer kapasiteye sahip bir “ikiz” QNN modeli oluşturularak, salt model farklılıkları yerine gerçekten **donanım kaynaklı enerji farklarına** odaklanılıyor. Sonuçta makale, *“Hangi algoritmik ve donanımsal koşullar altında SNN’ler uçtan uca enerji verimliliğinde ANN’leri gerçekten geride bırakır?”* sorusuna kapsamlı yanıtlar sunmaktadır.

Teknik Yöntemler ve Kullanılan Modeller

1. QNN–SNN Eşlenik Modeli: Çalışma, her bir SNN’ye karşılık gelen bir QNN modeli tanımlayarak adil bir karşılaştırma yapar. Bu eşlenik çiftlerde (ikiz modellerde) ağ yapısı ve ağırlık değer formatı birebir aynı olup sadece **aktivasyonların temsili** farklıdır [4]. Örneğin, T zaman adımı üzerinden çalışan bir SNN’ye karşılık, aynı yapıda ve ağırlıklarda fakat aktivasyonları $\lceil \log_2(T+1)\rceil$ bit ile kuantize edilmiş bir QNN oluşturulur [5]. Bu sayede her iki model de benzer temsil gücüne ve donanım gereksinimine sahip olur; karşılaştırma yapıldığında enerji farklarının sadece hesaplama tarzından (sürekli vs. zamana yayılmış) ve donanım kullanımından kaynaklanması sağlanır [6] [7]. Ayrıca Theorem 1 ve Theorem 2 ile formel olarak gösterilir ki böyle bir SNN-QNN çifti, tek bir nöronun çıkış düzeyleri ve seyreklik oranı bakımından denk bilgi kapasitesine sahiptir: SNN’nin T zaman adımındaki maksimum farklı çıktı sayısı QNN’nin $\log_2(T+1)$ seviyeli aktivasyonuyla eşleşir ve SNN’deki ortalama spike oranı s_r, QNN’deki aktivasyonların **seyreklik oranıyla** doğrudan ilişkilidir [8] [9]. (Theorem 2, $s_r^{worst} = 1-\gamma$ formülüyle SNN’deki en yüksek ortalama ateşleme oranının QNN’deki γ seyreklik oranına bağlı olduğunu ifade etmektedir.)

2. Analitik Enerji Modeli: Yazarlar, hem SNN hem de QNN (ANN) için donanım düzeyinde ayrıntılı bir **enerji tüketim modeli** geliştirmişlerdir [10]. Bu model, **hesaplama birimlerinin enerjisi** ile **veri**

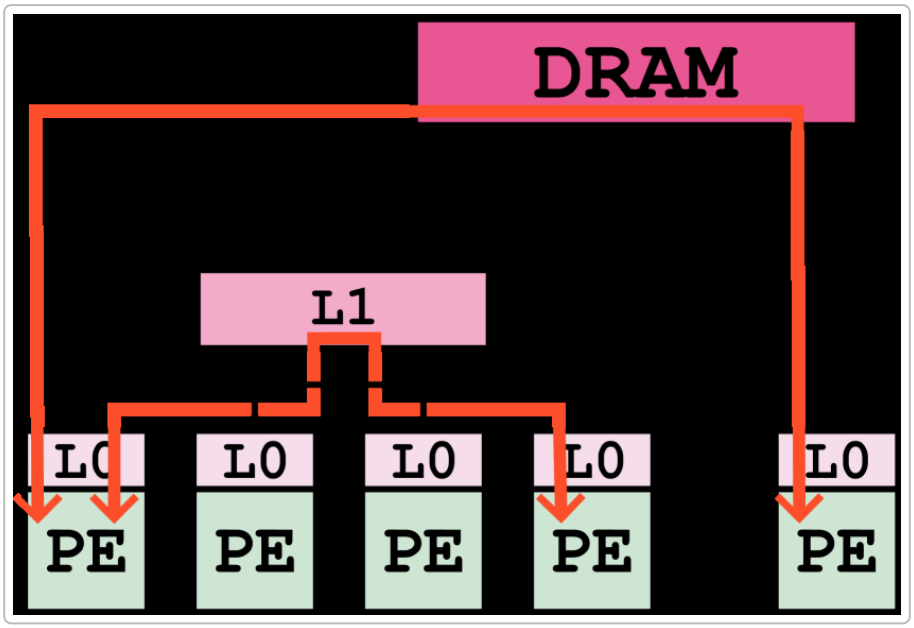
taşıma/bellek enerjisi bileşenlerini ayrı ayrı ele alır [11]. Toplam enerji, formülasyon olarak $E = E_{\text{Compute}} + E_{\text{Data}} (+ E_{\text{Control}})$ şeklinde ifade edilmiştir [11]. Burada:

- *Hesaplama Enerjisi (E_{Compute})* – SNN'lerde toplama işlemlerine (ve çok basit eşik kontrolüne), ANN'lerde ise çarpma-toplama (MAC) işlemlerine ve aktivasyon fonksiyonlarına harcanan enerji olarak tanımlanır. SNN'ler çarpma işlemlerini toplama ile ikame ederek teorik olarak avantaj sağlasa da, gerçek kazanç donanımdaki aritmetik birimlerin tasarımına, veri tiplerine ve çalışma esnasındaki **sparsity (seyreklik)** oranına bağlıdır [12].

- *Veri Taşıma Enerjisi (E_{Data})* – Bellek hiyerarşisinden hesaplama birimlerine veri getirme (ağırlıkların ve aktivasyonların taşınması) ve ara sonuçları ileri katmanlara iletme sırasında harcanan enerjidir [13]. Modern donanımlarda özellikle bellek erişimlerinin enerji bedeli çok yüksektir ve "*memory wall*" olarak bilinen etki nedeniyle işlemci hızları artsa da bellek erişimi enerji/verim sınırlayıcı olmaktadır [3] [14]. SNN'lerde her bir ağırlık değeri, T zaman adımı boyunca birden çok kez kullanılacağından (her bir nöron aktivasyonu için aynı ağırlık tekrar tekrar okunur), bellekten veri çağırma sıklığı ANN'lere kıyasla katlanır [15]. Örneğin, ANN her katmanda ağırlıkları bir kez yükleyip kullanırken, SNN eşleniğinde aynı ağırlık $T \times s_r$ kez (zaman adımı ve spike oranına bağlı olarak) kullanılacağından **bellek erişimi sayısı artar** [15]. Bu da eğer dikkat edilmezse SNN'nin toplam enerji tüketimini ciddi şekilde yükseltebilir.

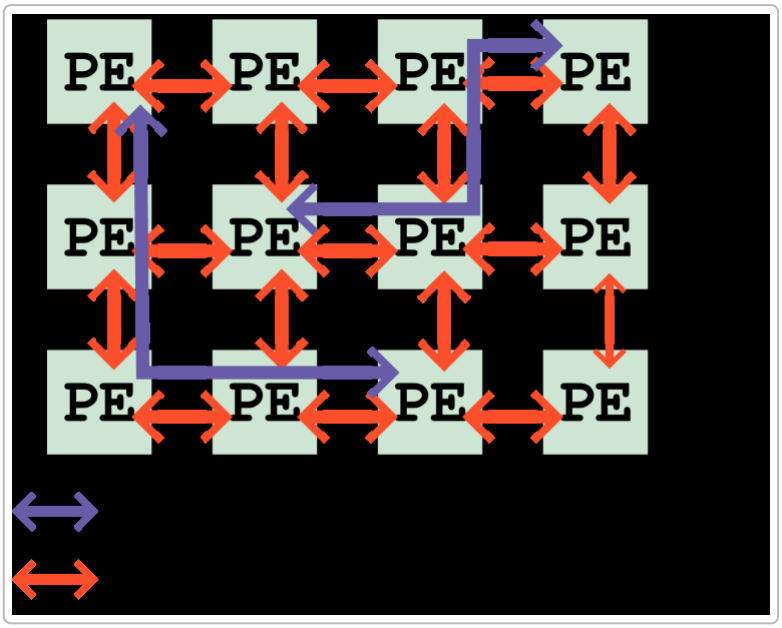
- *Kontrol Enerjisi (E_{Control})* – Donanımın getirdiği ek kontrol maliyetlerini (ör. işlemci ön yüzündeki komut getir & çözme ya da CPU/GPU'lardaki çizelgeleyiciler vb.) kapsar. Ancak makale, adil bir karşılaştırma için bu genel giderleri sabit varsayarak veya ihmal ederek asıl odak noktası olan **hesaplama ve veri aktarımı maliyetlerine** yoğunlaşmaktadır [16].

3. Donanım Mimarileri ve Veri-Yeniden Kullanımı: Çalışma, iki temel donanım mimarisi senaryosunu analiz ediyor: **klasik çok seviyeli bellek mimarisi** (örneğin GPU/TPU tarzı hızlandırıcılar) ve **dağıtık veri-akışlı mimari** (örneğin nöromorfik çipler veya NoC tabanlı özel donanımlar). Klasik mimaride hesaplama birimleri (ör. işlem çekirdekleri) altında L1, L0 önbellekleri ve en altta DRAM gibi bir **bellek hiyerarşisi** bulunur [17] [18]. Bu yapıda her katmanda **ağırlıklar önce DRAM'den SRAM'a yüklenir**, sonra aynı ağırlıklar birçok hesaplamada kullanılarak bellek erişimi sayısı azaltılır (ağırlıkların tekrar kullanımı) [18]. Yazarlar **"reuse factor" (yeniden kullanım faktörü)** denilen bir metriği tanıtarak, bir verinin yerel belleğe (SRAM) yüklendikten sonra kaç kez kullanıldığını tanımlamışlardır [19]. Özellikle evrişimli katmanlar için bu reuse factor, çekirdek boyutu, giriş haritası boyutu, adım ve dolgu miktarı gibi parametrelerle formül (Denklem 4) üzerinden verilmektedir [20]. Bu sayede ANN tarafında **ağırlık yeniden kullanımı** maksimize edilerek bellek enerjisi azaltılmaya çalışılır; SNN tarafında ise T adımlar boyunca aynı ağırlığı kullanabilmek için yeterli **yerel bellek kapasitesi** gereklidir (ağırlıklar tüm zaman penceresi için SRAM'de tutulur) [21] [22].



Şekil 1(a): Klasik bir hızlandırıcı mimaride, ana bellek (DRAM) ve seviye 1 önbelleğinden (L1) işlem elemanlarına (PE) veri transferleri gerçekleşir. Bu çok kademeli bellek erişimleri, hesaplama kadar (hatta çoğunlukla ondan daha fazla) enerji harcar. Yatay ve dikey oklar veri hareketini, pembe bloklar bellek katmanlarını, yeşil bloklar işlem birimlerini göstermektedir [18].

Dağıtık **veri-akışlı mimaride** ise büyük bir önbellek yerine **çok sayıda yerel bellekli işlem çekirdeği (PE)** ve bunları bağlayan bir **çip-üstü ağ (Network-on-Chip, NoC)** bulunur [23] [24]. Nöromorfik çipler bu kategoridedir: her bir PE, kendi yerel ağırlık belleğine sahiptir ve yeni bir spike (ateşleme sinyali) geldiğinde tetiklenerek hesaplama yapar, sonuç spike'ını ağ üzerinden ilgili diğer PE'ye iletir [25] [26]. Bu mimaride amaç, aktivasyonların belleğe yazılıp okunması yerine **router'lar üzerinden komşu çekirdeklere taşınmasıyla** enerji tasarrufu sağlamaktır [27] [26]. SNN'lerin yüksek düzeyde seyreklik içeren "spike" verisi, NoC üzerinde taşınırken klasik belleğe kıyasla daha verimli olabilir (az bit ile çok anlam taşındığı için) [27] [28]. Ancak mevcut dijital nöromorfik çipler genellikle sadece SNN için optimize edildiğinden, bu mimaride ANN'leri verimli çalıştırmak zor olabilir; makalede bu nedenle veri-akışlı mimaride sadece SNN çalıştırılması senaryosu incelenmiş ve **SNN'nin enerji formülü** ilgili parametrelerle (ör. ortalama *hop* sayısı, her bir router geçişinin enerji bedeli vs.) Denklem 6 olarak sunulmuştur [29] [30].



Şekil 1(b): Dağıtık veri-akışlı mimaride işlem birimleri (PE) arası iletişim bir ağ yapısı ile sağlanır. Kırmızı oklar komşu PElere doğrudan veri aktarım bağlantılarını, mor ok ise bir spike paketinin ortalama birkaç "hop" üzerinden hedefine ulaşmasını temsil ediyor. Bu mimari, uzun mesafe bellek erişimleri yerine yerel iletişimi önceliklendirdiği için SNN gibi seyrek etkinlik gösteren ağlar için enerji kazancı vaat eder [27] [26].

4. Enerji Denklemleri ve Kriterler: Her iki mimari için de yazarlar, SNN ve ANN'in enerji tüketimini formüllerle ifade etmişlerdir. **Klasik mimari** için, ANN modelinin bir katmandaki enerji maliyeti Denklem (3) ile verilmiştir [18] [31]. Bu denklemde her bir hedef nöron için, o nörona bağlı N_{src} kadar kaynak nöronun bulunduğu varsayılır; N_{src} sayısı ve ANN'in yararlanabildiği seyreklik oranı (γ) dikkate alınarak hesaplama ve bellek erişimi maliyetleri hesaplanır [31] [20]. SNN için ise aynı sistemde bir çıktı üretmenin maliyeti, zaman boyutunu da hesaba katarak Denklem (5) ile ifade edilir [32] [33]. Bu formülde SNN'de çarpımların yerine geçen toplama işlemlerinin daha düşük enerji gerektirdiği varsayımı, *ancak* T adım boyunca ağırlıkların SRAM'de tutulmasının getirdiği ek bellek maliyeti ile dengelenir [34] [22]. Ayrıca SNN'de her zaman adımında sadece 1-bit'lik girişler işlendiği için veri hareketinin tane bazında daha küçük paketlerle olduğu, ancak çok sayıda küçük paketin de verimsizlik yaratabileceği modele dahil edilmiştir [35] [36]. **Nöromorfik (veri-akışlı) mimari** için ise enerji denklemi (Denklem 6), klasik mimariden farklı olarak DRAM yerine NoC üzerindeki *router* geçiş sayılarını (hop) ve her bir hop için harcanan enerjiyi içerir [29]. Örneğin, bir spike paketinin hedef PEye ulaşmak için ortalama h adet ara noktadan geçtiği varsayılarak toplam taşıma enerjisi $h \times E_{\text{router}}$ şeklinde modele eklenmiştir [30]. Bu mimaride ANN çalıştırmak genellikle verimsiz olduğundan, yazarlar Denklem 6'yı sadece SNN için verip ANN ile kıyaslamayı dolaylı yapıyorlar; ama genel olarak SNN'nin NoC tabanlı mimarideki avantajı, klasik mimaride bellek erişimine harcanacak enerjinin önemli bir kısmını iletişim ağı üzerinde daha ucuza yapabilmesinden geliyor.

5. Eğitim Stratejisi ve Seyrekleştirici Düzenlileştirme: Makale sadece teorik analiz yapmakla kalmayıp, SNN'lerin enerji verimliliğini artırmak için yeni bir eğitim yaklaşımı da öneriyor. Yazarlar, SNN'lerin ANN'leri geçebilmesi için **çok yüksek oranlarda seyreklik** gerektiğini tespit ettikten sonra (aşağıda deneysel sonuçlarda detaylandırıldı), bu seyrekliği arttırmak amacıyla eğitim aşamasında iki farklı **düzenlileştirme (regularization) terimi** ekliyorlar [37] [38]. Çalışmada SNN'ler doğrudan eğitilmemiş, bunun yerine önce normal bir ANN eğitilip sonradan SNN'ye dönüştürülmüştür (ANN-to-SNN dönüşümü) [39] [40]. Dönüşüm esnasında doğruluk kaybını azaltmak için, ANN eğitiminde ReLU aktivasyonları yerine SNN'lerin ikili çıktısına daha yakın davranan "clamp" (belirli aralıkta sınırlama) ve

kuantizasyon fonksiyonları kullanılmıştır (Denklem 1 ve 2) [40] [41]. Bu sayede ANN'de aktivasyonlar 0-1 aralığına çekilip T adımda ateşlemeye uygun şekilde kuantize edilerek SNN ile uyumlu hale getirilir.

Daha sonra, **seyrekliği artırmak** için iki ilave düzenlileştirme yaklaşımı uygulanmıştır:

- *Ağırlık Küçültme Regularizasyonu:* Ağırlıkların küçülmesinin, nöronların zamana yayılmış toplam uyarılma miktarını düşürerek daha az spike üretmesine yol açacağı fikrine dayanır [42]. Bu amaçla kayıp fonksiyonuna, tüm ağın ağırlıklarının L2 normunu cezalandıran bir terim eklenmiştir (Denklem 11 ve 12) [42]. Bu terim bir λ katsayısı ile ölçeklendirilerek, ağırlıkların büyüklüğünü sınırlandırma derecesi ayarlanır. Uygulamada farklı λ değerleri denenmiş ve $\lambda=0.05$ civarında iyi bir denge bulunduğu belirtilmiştir (çok büyük λ doğruluğu düşürürken, çok küçük λ ise kayda değer sparsity artışı sağlamıyor) [43] [44]. Örneğin, $\lambda=0.05$ ile CIFAR-10 üzerindeki VGG16 ağında doğruluk yaklaşık %93 seviyesinde korunurken seyreklik oranı %92.9 olmuştur; $\lambda=0.1$'e çıkarıldığında doğruluk %1'den fazla düştüğü gözlemlenmiştir [43] [44].

- *Aktivasyon Kısıtlama Regularizasyonu:* İkinci strateji, aktivasyon seviyelerini doğrudan düşürmeye yöneliktir. ANN eğitimi sırasında her katmanın aktivasyonları 0 ile 1 arasında *clamp* edilip kuantize edildikten sonra, bu değerlerin toplamını cezalandıran bir terim kayba eklenir [45]. Yani ağ genelinde aktivasyonların mümkün olduğunca düşük seviyede kalması teşvik edilir (Denklem 13 ve 14) [45]. Bu terim için de bir ölçek faktörü (ör. η) kullanılarak, ne derece güçlü uygulanacağı kontrol edilir. Yazarlar η için 10^{-6} değerinin en iyi sonuçları verdiğini raporlamışlardır: daha büyük (10^{-5}) bir değer aşırı kısıp doğruluğu %86'ya düşürürken, daha küçük (10^{-7} gibi) değerler ise sparsity kazanımını azaltmaktadır [46]. Sonuçta $\eta=10^{-6}$ ile eğitim yapıldığında, model hem yüksek doğruluğunu koruyup hem de seyrekliği ciddi oranda artırabilmiştir [46].

Bu iki düzenlileştirmenin birlikte uygulanması, SNN modelinin **ateşleme oranını ciddi ölçüde azaltmış** ve dolayısıyla enerji tüketimini düşürmüştür. Makale, bu tekniklerin bir "sparsity-aware training algorithm" başlığı altında Şekil 6 ile görselleştirildiğini belirtmektedir [38] [47].

Deneysel Kurulum ve Sonuçların Sunumu

Makaledeki deneysel çalışma, geliştirilen teorik modellerin ve önerilen eğitim tekniklerinin **pratik etkilerini** değerlendirmek üzere yapılmıştır. Deneysel kurulum şu şekildedir:

- **Modeller ve Veri Kümeleri:** Popüler VGG ağ ailesi kullanılmış, özellikle VGG16 modeli üzerinde durulmuştur. Ayrıca daha küçük bir varyant (makalede VGG *olarak anılmış, muhtemelen VGG11 benzeri daha küçük bir yapı) ve VGG13, VGG19 gibi farklı boyutlarda ağlar da teste dahil edilmiştir* [48] [49]. *SNN ve eşleniği QNN modelleri, CIFAR-10 ve daha zor bir veri seti olan CIFAR-100** üzerinde değerlendirilmiştir [48] [50]. Bütün ağlar PyTorch kullanılarak eğitilmiş ve dönüştürülmüştür; geliştirilen çerçevenin kodu da kamuyla paylaşıldığı belirtiliyor.

- **Zaman Penceresi (T) Değeri:** Denemelerde SNN'lerin zaman adımı sayısı genellikle **$T=6$** olarak sabitlenmiştir (özellikle sonuç tablolarında $T=6$ görülmektedir) [51] [49]. Yazarlar, T'yi 4 ile 8 arasında değiştirerek de incelemeler yapmış ve T çok küçük olduğunda doğrulukta ciddi düşüşler, çok büyük olduğunda ise enerji maliyetinde artışlar gördüklerini aktarmışlardır [52]. $T=6$, doğruluk ve enerji arasında makul bir denge sunduğu için ana deneylerde tercih edilmiştir (örneğin $T=4$'ten 6'ya çıkarıldığında doğruluk %92.7'den %93.3'e yükselirken, $T=8$'e çıkınca doğruluk daha da artsa da enerji maliyeti de artmıştır) [52] [43].

- **Değerlendirme Ölçütleri:** SNN ve ANN modellerini karşılaştırmak için **doğruluk (accuracy)**, **seyreklik oranı (sparsity %)** ve en önemlisi **enerji tüketim oranı** kullanılmıştır. Enerji karşılaştırmaları, analitik modelden elde edilen hesaplamalara dayanmaktadır: Her bir modelin hem klasik mimaride hem de veri-akışlı (nöromorfik) mimaride harcayacağı toplam enerji tahmin edilerek oranlanmıştır. Sonuçlar genellikle **"SNN enerjisi / ANN enerjisi"** biçiminde bir oransal değer olarak sunulmuştur [49]. Örneğin 0.85 değeri, SNN'nin ANN'nin %85'i kadar enerji harcadığını (yani %15 tasarruf sağladığını) ifade eder.

- **Sonuçların Sunumu:** Makalede bulgular çeşitli tablolar ve şekillerle sunulmuştur. **Tablo 5**, CIFAR-10 ve CIFAR-100 sonuçlarını özetlemekte; burada her ağ için SNN doğruluğu, seyreklik oranı ve enerji oranları verilmektedir [53] [54]. **Şekil 7 ve Şekil 8**, doğruluk-seyreklik ve doğruluk-enerji arasındaki trade-off'ları görselleştirmektedir (örneğin T arttıkça spike sayısının ve dolayısıyla enerji tüketiminin nasıl yükseldiği, fakat doğruluğun da arttığı gösteriliyor) [52] [55]. Ayrıca **Şekil 9**, belirli koşullarda SNN ile ANN enerjisinin eşitlendiği noktaları (break-even noktalarını) grafiksel olarak sunarak hangi T ve s_r değerlerinin kritik olduğunu ortaya koymuştur [30].

Önemli Bulgular

- **Seyreklik ve T Koşulları:** Makalenin belki de en çarpıcı bulgusu, SNN'lerin ANN'lerden gerçekten daha az enerji tüketebilmesi için **çok katı koşulların sağlanması gerektiğidir**. VGG16 modeli üzerinde yapılan analizlerde, **T ne kadar büyükse SNN'nin enerji avantajı elde etmesi için o kadar aşırı bir seyreklik gerektiği** görülmüştür. Örneğin, T oldukça büyük (makalede net değer verilmese de tipik bir üst sınır olarak 16 veya daha fazla düşünebiliriz) ise SNN'nin >%97 gibi ekstrem bir nöron seyrekliğine sahip olması zorunludur; aksi takdirde ANN enerji verimliliğini geçemiyor. Daha makul bir zaman penceresi olan **$T=6$** için bile SNN'nin **>%93** seyreklik oranına ulaşması gerektiği belirtilmiştir. Bu oranlar, pratikte çoğu SNN'nin erişemeyeceği kadar yüksektir. Nitekim yazarlar mevcut literatürdeki tipik SNN'lerin bu *"çok yüksek sparsity & küçük T"* gereksinimlerini karşılayamadığını ve dolayısıyla iddia edilen enerji avantajlarının çoğunlukla gerçekleşmediğini vurgulamışlardır. Bu tespit, çalışmanın ana mesajlarından biridir: *SNN'ler ancak çok seyrek ateşleme yaparlarsa ve zaman penceresi oldukça sınırlıysa gerçekten enerji kazancı sağlarlar* [7] [56].

- **Tipik Koşullarda Eşik Değerler:** Makalede parametre uzayı taranarak spesifik eşik değerler ortaya konmuştur. Örneğin, *"tipik nöromorfik donanım koşulları altında"* (yani SNN'lere özel optimize edilmiş bir veri-akışlı çip senaryosunda), orta büyüklükte zaman pencerelerinde ($T \in [5, 10]$) SNN'nin **ortalama ateşleme oranı $s_r < %6.4$** olursa ancak QNN eşdeğerinden daha verimli olacağı hesaplanmıştır [7] [56]. Bu, çok düşük bir etkinlik oranıdır (yani nöronların %93.6'dan fazlası sessiz kalmalı). Böyle somut sayısal rehberler, SNN tasarlayanlara sistemin ne kadar seyrek çalışması gerektiğini gösteriyor.

- **Eğitimle Elde Edilen Sonuçlar:** Önerilen düzenlileştirme stratejileri uygulandıktan sonra, yazarlar SNN'lerin belirtilen zorlu kriterlere yaklaşabildiğini göstermişlerdir. CIFAR-10 üzerinde $T=6$ ile eğitilen **VGG16 tabanlı SNN**, yaklaşık **%92.76 doğruluk** ve **%94.19 seyreklik** elde etmiştir [48] [46]. Bu SNN modelinin enerji tüketimi, eşlenik ANN modelinin enerjisinin sadece **%85'i kadarı**dır (klasik mimaride) ve **%78'i kadarı**dır (özel veri-akışlı mimaride) [49]. Başka bir deyişle, optimizasyonlarla donatılmış SNN, klasik donanımda ANN'den %15, özel donanımda ise %22 daha az enerji harcamıştır. Daha küçük bir VGG versiyonu olan "VGG*" modelinde enerji kazancı daha da yüksektir (CIFAR-10'da SNN enerji oranı 0.70/0.69, yani %30 civarı tasarruf) [49]. CIFAR-100 sonuçlarında da benzer eğilimler görülmüş, ancak beklendiği üzere doğruluk seviyeleri daha düşük kalmıştır (ör. VGG16 SNN için %69.4 doğruluk, %93.98 sparsity) [50] [57].

- **Karşılaştırma ve Önceki Çalışmalarla Sonuçlar:** Son bölümde, yazarlar elde ettikleri enerji verimlilik sonuçlarını önceki çalışmalardaki SNN başarımlarıyla kıyaslamışlardır [58]. Örneğin, kendi SNN modelleri %94 gibi yüksek doğruluklarda çalışırken enerji tüketimini ciddi oranda azalttığı için literatürdeki diğer SNN uygulamalarından daha ileri bir enerji-verimlilik dengesi yakaladıklarını belirtirler. Ayrıca, ANN vs SNN konusunda benzer analiz yapan araştırmaların genellikle bellek maliyetlerini tam hesaba katmadığını, bu nedenle SNN'lere aşırı iyimser yaklaştığını hatırlatıp, **kendi çalışmalarının bu boşluğu doldurduğunu** ifade ediyorlar.

Matematiksel Tanımlar ve Formüller

Makale, içerdiği teknik detaylar açısından oldukça zengin olup pek çok denklem, tanım ve teorem sunmaktadır. Aşağıda önemli matematiksel unsurlar özetlenmiştir:

- **SNN-QNN Eşdeğerlik Teoremleri:** *Theorem 1*, T zaman adımlı bir ateşlemeli sinir ağının, aktivasyonları $\lceil \log_2(T+1)\rceil$ bit çözünürlükte olan bir kantize ANN ile **aynı çıktı temsil kapasitesine** sahip olduğunu kanıtlar [4] [5]. Bu, SNN'nin 0 ile T arasında üretebileceği farklı spike sayılarının (toplam ateşleme sayısı) bir ANN'in belli bit derinliğindeki aktivasyon değerlerine karşılık geldiği anlamına gelir. *Corollary* olarak da bu eşdeğerliğin pratikte geçerli olması için SNN'deki membran potansiyelinin her zaman sınırlı aralıkta kalması (taşmaması) gerektiği belirtilir – ki bu da normal çalışma koşullarında sağlanabilir. *Theorem 2* ise SNN'nin **ortalama ateşleme oranı (s_r) ile QNN'nin aktivasyon seyreklik oranı (γ)** arasındaki ilişkiyi formülle ortaya koyar [8]. Basitçe ifade etmek gerekirse, QNN'de aktivasyonların γ oranında sıfır olması demek, SNN'nin en kötü durumda $s_r = 1-\gamma$ oranında spike üretebileceğini gösterir [9]. En iyi durumda ise (spike'ların zaman içinde dağıtılmasıyla) SNN ateşleme oranı $s_r = \frac{1-\gamma}{T}$ mertebesine kadar düşebilir [59]. Bu teorem, ANN'deki seyrekliğin SNN'ye ne kazandırabileceğini veya sınırlarını matematiksel olarak tanımlamaktadır.

- **Enerji Hesaplama Formülleri:** Makalede farklı mimariler için enerji formülleri verilmiştir. Genel olarak toplam enerji **$E_{\text{total}} = E_{\text{Compute}} + E_{\text{Data}} (+ E_{\text{Control}})$** şeklinde ayrıştırılır [11]. *Denklem (3)*, klasik mimaride bir ANN katmanının enerji tüketimini ifade eder [18]. Bu formülde her bir hedef nöron için gerekli olan DRAM→SRAM veri transfer enerjisi (ağırlıkların getirilmesi) ve yerel SRAM→hesaplama enerjisi, N_{src} (nöronun giriş bağlantı sayısı) ve γ (ANN'nin girdi aktivasyonlarındaki seyreklik) cinsinden ifade edilir [31]. *Denklem (4)*, özellikle evrişimli katmanlar için **yeniden kullanım faktörünü (RF)** hesaplar: Bir giriş özelliğinin yerel belleğe yüklendikten sonra ortalama kaç kez kullanıldığını, dolayısıyla kaç çarpma/toplamada tekrar harcandığını gösterir [20]. Bu formül giriş haritasının yüksekliği, genişliği, filtre boyutu, adım (stride) ve padding gibi parametrelerle verilir. *Denklem (5)*, klasik mimaride bir SNN'nin enerji tüketim formülüdür [32]. Bu formül, ANN'nin Denklem 3'üne benzer bir yapıda olup, çarpma yerine toplama işlemlerinin daha düşük enerji maliyetiyle, fakat T adım boyunca yinelenen hesaplamaların ve artan bellek kullanımının getirdiği maliyetle dengelendiği bir ifadedir [22] [33]. Özellikle SNN formülünde, her bir zaman adımında 1-bit'lik veri transferinin verimsizliği ve ağırlıkları T boyunca SRAM'de tutmanın gerektirdiği ek enerji terimleri bulunur [35] [60]. *Denklem (6)* ise nöromorfik veri-akışlı mimaride SNN için enerji hesabıdır [29]. Bu denklemde NoC üzerindeki hareketin enerji bedeli, örneğin bir spike'ın ortalama $\bar{h}$ adet yönlendiriciden geçmesi durumunda $E_{\text{NoC}} = \bar{h} \cdot E_{\text{hop}}$ gibi bir terim olarak eklenir (burada E_{hop} tek bir router geçişinin enerji maliyetidir) [30]. Yazarlar şekil üzerinde de bunun ortalama 6 atlamalı bir yol ile gösterildiğini belirtiyorlar. Bu formüller, farklı senaryolarda (ör. ağırlıkların hiç yeniden

kullanılmadığı en kötü hal vs. tam kullanıldığı en iyi hal) parametre değerleri yerine konarak analizler yapılmıştır.

- **Düzenlileştirme Terimleri:** Eğitim esnasında kullanılan regularization terimleri de formüllerle tanımlanmıştır. *Denklem (11)* ve *(12)*, ağırlıklar için eklenen L2 cezalandırma terimini göstermektedir [42]. Temel çapraz-entrofi kayıp fonksiyonuna $\lambda \cdot |W|_2^2$ şeklinde bir ekleme yapıldığı söylenebilir (tam formül metinde verilmemiş olsa da, L2 normun λ ile ölçeklenmesi şeklinde) [42]. *Denklem (13)* ve *(14)* ise aktivasyonları küçültmek için eklenen terimi tanımlar [45]. Bu da kabaca $\eta \sum \text{Act}$ gibi (ağdaki tüm aktivasyon değerlerinin toplamını η katsayısıyla çarpıp kayba eklemek) bir formüle denktir. Her iki düzenlileştirme de eğitimde geri yayılım sırasında ilgili türevleri alınıp ağ parametrelerini seyrekliliği artıracak yöne doğru itmektedir.

- **Teknik Tanımlar:** Makale boyunca geçen bazı önemli tanımları da vurgulayalım: **Spike rate (s_r)** – Bir SNN'de bir nöronun belirli bir zaman penceresinde ortalama ateşleme sıklığı, toplam T adımdaki 1 değerlerinin oranı olarak tanımlanabilir. **Sparsity (seyreklik) oranı (γ)** – Bir katmanda veya ağ genelinde aktivasyonların ne kadarının sıfır olduğunu ifade eder (1-γ ise etkin aktivasyon oranıdır). **Membran potansiyeli (V)** – SNN'de nöronun biriktirdiği değer; her adımda gelen girdilerle artar ve eşik (θ) seviyesini aşarsa nöron "ateşler" ve V resetlenir [61]. **Integrate-and-fire modeli** – SNN'lerde en yaygın nöron modeli olup, her zaman adımında $V \leftarrow V + \sum w_i x_i$ şeklinde girdi biriktiren, eşiği aşınca çıkış veren mekanizmadır [62] [63]. Makalede Şekil 1'de bu modelin bir görseli sunulmuş, ikili spike trenleri ve eşik kontrolü örneklenmiştir. **NoC (Network-on-Chip)** – Çip üzerindeki yönlendirici ağı; PElere gelen spike paketleri yönlendiriciler arasında iletilir. **Reuse Factor (RF)** – Bir verinin (ör. bir ağırlık veya aktivasyon bloğunun) SRAM'e alındıktan sonra kaç hesaplamada kullanıldığını gösteren sayı; ne kadar büyükse bellek erişimi o kadar verimli kullanılıyor demektir [64] [19].

Uygulama Alanları ve Potansiyel Etkiler

Bu çalışmanın bulguları ve önerileri, **enerji kısıtlı yapay zeka uygulamaları** için doğrudan önem taşımaktadır. Özellikle şu alan ve durumlarda etkili olması beklenir:

- **Kenar Bilişim (Edge Computing) ve Gömülü Sistemler:** Pille çalışan cihazlar, IoT sensör ağları, giyilebilir teknolojiler gibi ortamlarda enerji verimliliği birincil önceliktir. SNN'ler, teoride bu ortamlara uygun görülüyordu çünkü seyrek ve etkin olduğunda daha az enerji harcayabilirler. Bu çalışma, SNN'lerin gerçekten ne zaman avantaj sağlayacağını netleştirdiği için, kenar cihazlarda **hangi tip yapay zeka modelinin seçileceği** konusunda yol gösterici olacaktır [14] [65]. Örneğin, eğer uygulama sırasında sinyaller doğası gereği seyrek ise (belli olaylar nadiren tetikleniyorsa), SNN mimarileri tercih edilerek enerji tasarrufu elde edilebilir. Ancak sinyaller görece yoğun veya sürekli ise, benzer bir QNN modelinin daha uygun olacağı anlaşılır. Bu sayede mühendisler, uygulamalarına uygun doğru mimariyi seçip sistem tasarımını buna göre yapabilir.

- **Nöromorfik Donanım Geliştirme:** Çalışma, yeni nesil nöromorfik çiplerin tasarımına da ışık tutuyor. Yazarların analizine göre, nöromorfik veri-akışlı mimariler SNN'ler için avantajlı olmakla birlikte, bunlarda bile belirli eşik değerler gerekli [7]. Donanım tasarımcıları, örneğin **yerel bellek boyutlarını yeterli tutarak ağırlık reuse factorünü maksimize etmeye** ya da **NoC topolojisini optimize ederek** spike iletim mesafelerini azaltmaya önem vermelidir. Makalenin önerdiği **"spatial-dataflow" mimarisi** (komşu PElere öncelik veren mesh ağı) tam da bu amaçla geliştirilen yenilikçi bir yaklaşım olarak sunuluyor [66] [67]. Bu mimaride uzak mesafe veri

transferleri yerine komşular arası iletişim teşvik edilerek enerji tüketimi düşürülüyor. Gelecekteki nöromorfik çipler, bu tür topolojileri ve adaptif veri yollarını kullanarak SNN'lerin verimini artırabilir.

- **Derin Öğrenme ve Ağ Tasarımı:** Yazılım tarafında, araştırmacılar artık SNN tasarlarken **seyreklik** kavramına çok daha fazla odaklanmak zorunda kalacaklar. Bu çalışma gösteriyor ki seyreklik, SNN'lerin can damarı – yüksek sparsity yoksa enerji avantajı da yok. Bu nedenle, SNN'ler için yeni eğitim teknikleri, düzenlileştirme yöntemleri (örneğin bu makalede sunulan ağırlık ve aktivasyon ceza terimleri gibi) geliştirerek ağları olabildiğince seyrek ateşleme yapar hale getirmek önemli. Bu yaklaşım, **model sıkıştırma** ve **verimlilik optimizasyonu** çalışmalarına da paralel bir katkı sağlar. Hatta sadece SNN değil, klasik ANN'lerde bile enerji verimliliği için bu makalenin vurguladığı unsurlar (seyrek aktivasyon, ağırlık paylaşımı vs.) uygulanabilir. Örneğin, bir **QNN modelini tasarlarken** bu çalışmanın QNN-SNN eşdeğerliği sayesinde, eğer beklenen aktivasyon seyrekliği y ise o durumda SNN'nin ortalama s_r değeriyle nasıl bir trade-off olduğu anlaşılabilir; belki de QNN tarafında mimari kararlar buna göre alınabilir.

- **Akademik ve Endüstriyel Ar-Ge'ye Etkisi:** Bu çalışma, SNN'lerin enerji verimliliği iddialarına daha dikkatli ve niceliksel bakılması gerektiğini ortaya koyarak alandaki bazı yanlış kanıları düzeltme potansiyeline sahiptir. Enerji verimliliği, yapay zeka donanımı geliştiren endüstri için kritik bir metriktir (örneğin otonom araçlarda, drone'larda veya büyük veri merkezlerinde bile enerji tasarrufu önemlidir). Makalenin sağladığı analiz çerçevesi, **yapay zeka hızlandırıcılarının tasarımında** (hem donanım hem yazılım optimizasyonu tarafında) rehberlik edebilir. Örneğin, firma X bir SNN tabanlı yonga tasarlarken, bu çalışma sayesinde ürünün hangi uygulamalarda gerçekten fark yaratacağını, hangi uygulamalarda ise klasik ANN'lere yenik düşebileceğini önceden hesaplayabilir. Dolayısıyla Ar-Ge yatırımları daha bilinçli yönlendirilebilir.

Özetle, makalenin pratik etkisi SNN konseptini kullanan tüm **enerji-duyarlı yapay zeka uygulamalarını** kapsar: Doğru koşullarda SNN kullanmak büyük tasarruflar sağlarken, yanlış koşullarda beklentiye girmemek gerektiği anlaşılmıştır.

Benzer Çalışmalar ve Makalenin Yenilikçi Katkıları

Bu çalışmayı alandaki önceki işler ile kıyasladığımızda belirgin birkaç yenilik ve katkı öne çıkmaktadır:

- **Kapsamlı ve Adil Enerji Analizi:** Daha önceki pek çok çalışma, SNN'lerin enerji avantajını gösterirken genellikle sadece hesaplama adımlarını saymak (toplamalar vs çarpmalar) gibi basitleştirilmiş metrikler kullanıyordu [2]. Bellek erişimlerinin enerji maliyeti ve donanım detayları çoğunlukla ihmal ediliyordu. Bu makale ise **uçtan uca (end-to-end) enerji değerlendirmesi** yaparak büyük bir boşluğu doldurmuştur. Hem hesaplama hem de veri hareketi giderlerini içeren analitik bir model sunulmuş, böylece SNN vs ANN karşılaştırması gerçekçi bir temele oturtulmuştur [3] [10]. Özellikle ANN ve SNN'nin *aynı kapasitede* yapılandırılması (QNN-SNN ikizleri) sayesinde "elma ile elma" kıyaslaması mümkün olmuştur. Bu yaklaşım, benzer çalışmalarda pek görülmeyen ve oldukça titiz bir metodolojidir.

- **Teorik Eşdeğerlik ve Kriterler:** Makale, SNN ve ANN'lerin temsil gücünü ve sparsity ilişkisini **teorik teoremlerle** ortaya koyarak yenilikçi bir bakış sunuyor. Theorem 1 ve Theorem 2 ile ilk kez SNN zaman boyutu – ANN bit derinliği denkleştirilmiş ve ateşleme oranı – aktivasyon oranı ilişkisi kurulmuştur. Bu, literatürde SNN'lerin kapasitesiyle ilgili önemli bir katkıdır. Ayrıca parametre uzayının taranmasıyla bulunan %6.4 gibi kritik eşiğin açıkça belirtilmesi, **niceliksel rehberlik**

sağlayan bir yeniliktir [7]. Önceki çalışmalar genelde niteliksel çıkarımlar yaparken, bu çalışma *"şu koşulda SNN avantajlıdır, bu koşulda değildir"* şeklinde net sınırlar çizmektedir.

- **Yeni Donanım Tasarım Önerisi:** Yazarlar, yalnızca mevcut mimarileri değerlendirmekle kalmamış, aynı zamanda **iyileştirilmiş bir mimari tasarım konsepti** de ortaya atmıştır. Özellikle *spatial-dataflow architecture* adıyla bahsedilen, komşu PElere öncelik veren mesh tabanlı NoC tasarımı, SNN'lerin veri iletişim maliyetini düşürmeye yönelik özgün bir fikirdir [68] [67]. Bu mimarinin nasıl enerji tasarrufu sağladığı analiz edilmiş ve nöromorfik sistemler için gelecekteki yönelimler arasında sayılabilecek bir yaklaşım olarak sunulmuştur. Böylece makale, sadece analiz değil **tasarım önerisi** de getirerek benzer işlerden ayrılmaktadır.

- **Eğitim Yöntemlerinde İyileştirme:** Literatürde SNN'lerin doğruluk kaybı yaşamadan çalışması için çeşitli eğitim teknikleri önerilmişti, ancak bu makale enerji verimliliğini doğrudan hedefleyen bir eğitim düzenlileştirmesi önermesiyle öne çıkıyor. Ağırlık ve aktivasyonlar için eklenen regularizer'lar, SNN'lerin sparsity'sini artırmakta oldukça başarılı olmuş ve ilk defa SNN'yi ANN'den belirgin şekilde daha verimli hale getirmiştir [46] [51]. Bu katkı, SNN araştırmalarında yeni bir yön açabilir: *"Enerji-farkındalıklı öğrenme"*. Benzer çalışmalar genellikle ya donanım tarafına odaklanır ya da salt doğruluk optimizasyonu yapardı; burada ise eğitim sürecine enerji hedefi entegre edilmiştir.

- **Deneysel Performans ve Devreye Alınabilirlik:** Yazarların elde ettiği sonuçlar, aynı doğruluk seviyesinde ANN'nin %70-85 enerji tüketimiyle çalışabilen SNN'ler olduğunu gösteriyor [49]. Bu, alanda bir **durum kanıtı (proof-of-concept)** niteliğindedir. Önceki işlerde ya SNN doğrulukları düşük kalıyor ya da enerji avantajı net gösterilemiyordu. Burada ise %94'e varan doğruluklarla, anlamlı enerji tasarrufu birlikte başarılmıştır. Bu başarım seviyesi, SNN'lerin pratikte kullanılabilirliğini artıran önemli bir katkıdır ve benzer çalışmalar arasında öne çıkar.

Sonuç olarak, *"Reconsidering the energy efficiency of spiking neural networks"* makalesi, SNN'lerin enerji verimliliği iddiasını tüm boyutlarıyla masaya yatıran ve hem teori hem pratikte önemli yenilikler getiren bir çalışmadır. Hem araştırmacılar hem de mühendisler için değerli içgörüler sunarak, gelecekte enerji-verimli sinir ağları tasarlama yolunda bir rehber niteliğindedir [69] [70]. Bu makale sayesinde, SNN'lerin ne zaman gerçekten avantaj sağlayacağı netleşmiş ve bu avantajı gerçekleştirmek için gereken yöntemler ortaya konmuştur. Böylelikle çalışma, benzerleriyle kıyaslandığında gerek metodoloji kapsamı, gerek elde edilen bulgular, gerekse pratik öneriler açısından kayda değer bir katkı sunmaktadır.

Kaynaklar: Makaledeki tüm teknik içerik ve sonuçlar, arXiv'de yayınlanan ilgili çalışmadan alınmış ve özetlenmiştir [6] [51].

---

[1] [2] [3] [4] [5] [6] [7] [8] [9] [10] [11] [12] [13] [14] [15] [16] [56] [59] [61] [62] [65] [69] [70] Reconsidering the energy efficiency of spiking neural networks
https://arxiv.org/pdf/2409.08290

[17] [18] [19] [20] [21] [22] [23] [24] [25] [26] [27] [28] [29] [30] [31] [32] [33] [34] [35] [36] [37] [38] [39] [40] [41] [42] [43] [44] [45] [46] [47] [48] [49] [50] [51] [52] [53] [54] [55] [57] [58] [60] [63] [64] [66] [67] [68] [2409.08290] Reconsidering the energy efficiency of spiking neural networks
https://ar5iv.org/pdf/2409.08290