

Yonga Üstü Heterojen Çok İşlemciler İçin Enerji Verimli İş Parçacığı Eşlemesi Task Scheduling On Heterogeneous Chip Multiprocessors for Reducing Energy

Süleyman Tosun, Yusuf Yavuz

Bilgisayar Mühendisliği Bölümü
Ankara Üniversitesi

tosun@eng.ankara.edu.tr, yyavuz@eng.ankara.edu.tr

Özet

Periyodik iş parçacıklarının çok işlemcili mimarilere performans, maliyet, enerji ve güvenilirlik kısıtları göz önüne alınarak zamanlaması gömülü sistemlerdeki en zorlu işlerden biridir. Bu makalede, verilen iş parçacığı kümesini heterojen çok işlemci mimarisine eşleyen tamsayı doğrusal programlama (TDP) tabanlı bir model sunulmaktadır. Bu model, birçok objektif için kullanılabilir: enerjii minimize etmek, maliyeti (heterojen işlemci sayısını) minimize etmek ve güvenilirliği maksimize etmek. Enerji tüketimini azaltmak için dinamik gerilim ölçeklemesi kullanılarak güvenilirliği artırmak için iş parçacıkları kopyalanmaktadır.

Abstract

Scheduling periodic tasks onto a multiprocessor architecture under several constraints such as performance, cost, energy, and reliability is a major challenge in embedded systems. In this paper, we present an ILP (Integer Linear Programming) based framework that maps a given task set onto an HCM (Heterogeneous Chip Multiprocessors) architecture. Our framework can be used with several objective functions; minimizing energy, minimizing cost (number of heterogeneous processors), and maximizing reliability. We use Dynamic Voltage Scaling (DVS) for reducing energy consumption while we employ task duplication to maximize reliability.

1. Giriş

Son yıllarda gömülü sistem uygulamalarının karmaşıklığı oldukça fazla artmıştır. Bunun sonucu olarak, tasarımcılar yeni gömülü sistem mimarilerine yönelmişlerdir. Yonga üstü heterojen çok işlemciler (YHÇİ) [1,2] bu arayışa bir örnek olup birden fazla işlemcinin bir yonga üzerine yerleştirildiği mimarilerdir. YHÇİ mimarileri tasarımcıya tasarım amaçlarını karşılaması için çok büyük imkânlar sağlar çünkü bir uygulamanın YHÇİ üzerine birçok eşlemesi mümkündür. Fakat avantaj olarak görünen bu özellik çok karmaşık uygulamalarda dezavantaja döner: Uygulama eşlemesi son derece karmaşık bir hal alır.

Bir gömülü sistem tasarlanırken, bazı tasarım parametreleri diğerlerinden daha önemli olmaktadır. Mesela, pille çalışacak tasarımlarda enerji tüketimi ve güç önemli etkenlerken kritik uygulamalarda (uçaklar, roket atarlar, sağlık cihazları gibi) güvenilirlik öne çıkmaktadır. Bunların yanı sıra, tasarımda karşılanması gereken yonga alanı,

performans gibi kısıtlar da mevcuttur. Tüm bu objektif ve kısıtlar göz önüne alındığında, bir tasarımcının gömülü bir sistemi yazılım yardımı olmadan tasarlayabilmesi olanaksız görünmektedir.

Daha önceki çalışmalar genellikle verilen gömülü sistem kodunun yürütme çevrimini azaltmak üzerine yoğunlaşmıştır. Enerji tüketiminin minimize edilmesi ve güvenilirliğin artırılması konularında da çalışmalar mevcuttur. Fakat bilimiz dâhilinde enerji, güvenilirlik ve maliyet konularının bir arada incelendiği bir çalışma yoktur.

Bu çalışmadaki amaç, verilen gerçek zamanlı gömülü sistem uygulamasının YHÇİ mimarisine eşlemesidir. Bu eşleme tamsayı doğrusal programlama (TDP) yöntemiyle gerçekleştirilmiş olup çoklu kısıtlamalar altında bir objektif fonksiyonun iyileştirilmesi şeklinde çalışır. Bu kısıtlamalar ve objektif fonksiyonlar enerji, maliyet, performans ve güvenilirlik fonksiyonlarıdır. Sunulan bu sistem genel olarak üç farklı amaç için kullanılabilir:

- Enerji ve güvenilirlik kısıtlamaları altında maliyeti minimize etmek,
- Performans ve güvenilirlik kısıtlamaları altında enerji tüketimini minimize etmek,
- Maliyet ve enerji kısıtlamaları altında güvenilirliği maksimize etmek.

Enerji tüketimini minimize etmek için, literatürde sıkça kullanılan dinamik gerilim ölçekleme (DGÖ) tekniği kullanılmıştır. Literatürde de sıkça kullanılan bu yöntemin [4-7] temel prensibine göre, bir iş parçacığının çalışma gerilimi azaltıldığı zaman enerji tüketimi de azalacak fakat buna bağlı olarak çalışma süresi artacaktır. Değişik gerilim değerleri altında bir iş parçacığına ait birden fazla enerji tüketimi ve çalışma zamanı değeri bulunacağından, performans kısıtları altında en uygun enerji değerine sahip gerilim değerinde çalışan sürüm seçilmektedir. Güvenirliğin artırılması için ise iş parçacıkları kopyalanmaktadır. Yani, bir iş parçacığı birden fazla sürüm olarak çalıştırılmakta, sonuçları böylelikle karşılaştırılarak hata tespiti yapılabilmektedir. Bu yöntem de güvenilirliğin artırılması için daha önce kullanılmış [8], etkili bir yöntemdir. Sunulan model, bir açık yazılım tamsayı programlama paketi lp_solve 5.0 [3] üzerinde gerçekleştirilmiştir.

Bu makalenin kalanı şu şekilde organize edilmiştir: Bir sonraki bölümde enerji ve güvenilirlik metrikleri ile uygulama modelleri ve hedef mimari anlatılmaktadır. Bölüm 3’de TDP tabanlı yöntemin formülasyonu sunulmaktadır. Bölüm 4 deneysel sonuçları vermekte ve sonuçlar Bölüm 5’te tartışılmaktadır.

2. Ön bilgiler

2.1. Dinamik Gerilim Ölçekleme

Dinamik gerilim ölçeklemesi (DGÖ), güç tüketimi optimizasyonunda en fazla kullanılan tekniklerden biridir. Bu tekniğin çekiciliği iş parçacıklarının düşük gerilimde çalıştığında daha az enerji tüketmesinden gelir. Bu tekniğin olumsuz yanı ise düşük gerilimde iş parçacıklarının çalışma zamanlarının artmasıdır. Farklı gerilim değerlerinde iş parçacıklarının enerji ve en uzun çalışma süresi (EUÇS) arasındaki ilişkiyi incelemek için güç ve enerji arasındaki matematiksel ifadelerin verilmesi gerekmektedir. CMOS devrelerin güç sarfiyatı aşağıdaki eşitlikle ifade edilebilir [9]:

$$P = C_L \cdot N_s \cdot v^2 \cdot f \quad (1)$$

Bu eşitlikte C_L devrenin çıkış yük kapasitesini, N_s bir saat darbesindeki anahtarlama sayısını, v besleme gerilimini ve f saat frekansını temsil eder. Bir iş parçacığı daha düşük gerilimde çalışırsa, onun EUÇS değeri aşağıdaki ifadeyle orantılı olarak artar:

$$t = \frac{C_L \cdot v}{k \cdot (v - v_t)^\alpha} \quad (2)$$

Bu ifadedeki k , kullanılan yonga teknolojiye bağlı olan bir katsayıdır; v_t , eşik gerilimini ve α , 1,2 ile 2 arasında değişen teknoloji katsayısını ifade eder. Bu çalışmada sayısal hesaplamalar için $\alpha = 2$ ve $v_t = 0,6V$ olarak kabul edilmiştir. Ayrıca, eğer bir iş parçacığının EUÇS değeri yüksek gerilim değerinde (v_h) biliniyorsa, onun düşük gerilim değerindeki (v_l) EUÇS değeri aşağıdaki formülle hesaplanır:

$$t_{v_l} = t_{v_h} \cdot \frac{v_l}{v_h} \cdot \left(\frac{v_h - v_t}{v_l - v_t} \right)^2 \quad (3)$$

Benzer şekilde, bir iş parçacığının yüksek gerilim altındaki enerji tüketimi biliniyorsa, düşük gerilimdeki enerji tüketimi bulunabilir. Bunun için aşağıdaki eşitlik kullanılır.

$$E_{v_l} = E_{v_h} \cdot \left(\frac{v_l}{v_h} \right)^2 \quad (4)$$

Bu çalışmada, her bir iş parçacığının her bir işlemcideki EUÇS ve enerji tüketim değerlerinin verildiği varsayılmaktadır. Diğer gerilimdeki değerler (3) ve (4) nolu eşitlikler kullanılarak bulunmaktadırlar.

2.2. Güvenilirliği Artırmak

Bu çalışmadaki ikinci önemli konu tasarımın güvenilirliğinin artırılmasıdır. Bunun için, iş parçacığı kopyalama yöntemi kullanılmaktadır. Kopyalanan iş parçacıkları aşlının birebir aynısı olabileceği gibi farklı gerilim değerinde ve/veya farklı işlemci üzerinde çalışan değişik bir versiyonu da olabilir. İş parçacığı kopyalandığı zaman, sonuçların doğruluğunu karşılaştırmak için denetleyicilerin konulması gerekmektedir. Eğer iki sonuç birbirinden farklı ise, sistem ya baştan başlar veya geriye dönüş yöntemiyle [8] daha önce kaydedilmiş güvenli bir konumdan devam eder.

Modelimizde performans ve enerji tüketimi gibi kısıtlamalar nedeniyle tüm iş parçacıkları kopyalanamayabilir. Bu nedenle, güvenilirlik metriği olarak kopyalanan iş parçacıklarının yüzdesi kullanılacaktır. Ayrıca, asıl iş parçacığı ve kopya versiyonu aynı işlemci üzerine eşlenmeyecektir. Bunun nedeni ise işlemcinin hatalı olma ihtimalidir. Eğer, iki kopya iş parçacığı aynı işlemci üzerine eşlenirse, hatalı olan işlemcide iki sonuç doğru çıkabilir. Bundan kaçınmak ve hatalı olan işlemcilerin tespit edilebilmesi için böyle bir zorlama kaçınılmazdır.

2.3. İş Parçacığı Modeli ve Hedef Mimari

Bu çalışmada [11]'de bahsedilen periyodik iş parçacığı modeli kullanılmaktadır. Bu modelde bir iş parçacığı kümesi $T = \{t_1, t_2, t_3, \dots, t_n\}$ ile ifade edilir. İş parçacıkları periyodik olup birbirinden bağımsızdırlar. Her bir iş parçacığının kendisine ayrılmış bir başlangıç zamanı (iş parçacığının çalışmaya başlayabileceği en erken zaman) ve bitiş zamanı (iş parçacığının çalışma süresini tamamlayabileceği en geç zaman) vardır.

Hedef mimari bu çalışmada önemli rol oynamaktadır çünkü mimarinin konfigürasyonu sonuç tasarımın enerji, maliyet, performans gibi değerlerini etkilemektedir. Mesela farklı haberleşme teknikleri farklı haberleşme zamanlamalarına neden olacaktır ki bu da performans değerini etkiler. Bu çalışmadaki hedef mimaride her bir işlemci kendi hafızasına sahipken işlemciler arası haberleşme veri yolu üzerinden sağlanmaktadır.

2.4. Sistem Varsayımları

Bu çalışmada aşağıdaki varsayımlar yapılmıştır. Gerçek zamanlı uygulamalar incelendiğinde, yapılan bu varsayımların uygunluğu görülebilir.

1. Bir gerilim seviyesinden diğerine geçerken harcanan enerji ve zaman kaybı ihmal edilebilir düzeydedir.
2. Bir işlemcinin çalışmaya başlaması anındaki anahtarlardan doğan enerji tüketimi ve zaman kaybı ihmal edilebilir. Sadece iş parçacıklarının harcadığı enerji ve zaman göz önüne alınmalıdır.
3. İş parçacıkları birbirinden bağımsızdırlar ve çalışma anında başka işlem tarafından kesilemezler.
4. Bir iş parçacığının her bir işlemci üzerindeki EUÇS ve enerji tüketimi değerleri bilinmektedir.
5. İşlemciler gerilim değerlerini dinamik olarak değiştirebilirler.

3. İş Parçacığı Eşleme

Bu bölümde, TDP temelli formülasyonlar sunulacaktır. Tablo 1'de, bundan sonraki formüllerde kullanılacak olan yazım ve semboller verilmiştir.

3.1. TDP Formülasyonu

TDP formülasyonunda ilk önce ikili tabanda bir değişken olan $b_{i,j,v}$ tanımlanır. Bu değişken iş parçacığı i 'nin işlemci j 'ye v gerilim değerinde eşlendiğini gösterir. Burada not edilmelidir ki, her bir iş parçacığı mutlaka bir işlemciye eşlenmelidir. Fakat kopyalarının hepsi bir işlemciye eşlenemeyebilir çünkü diğer kısıtlamalar (performans gibi) tüm iş parçacıklarının kopyalanmasına izin vermeyebilir.

Tablo 2: TDP formüllerinde kullanılan yazım ve semboller.

Yazım	Tanım
o	Maksimum gerilim seviyesi sayısı.
m	Maksimum işlemci sayısı.
$Pmax$	Tüm iş parçalarının en son bitiş zamanı.
t_{ch}	Denetleyicinin çalışma süresi.
$t_{i,j,v}$	İş parçası i 'nin j işlemcisinde v gerilimindeki EUÇSsi.
$E_{i,j,v}$	İş parçası i 'nin j işlemcisinde v gerilimindeki enerjisi.
E_{max}	Enerji kısıtlaması
$Rmin$	Güvenilirlik kısıtlaması
$b_{i,j,v}$	Eğer iş parçası i işlemci j 'ye v gerilimde eşlendiyse 1; değilse 0.
d_i	İş parçası i 'nin çalışma süresi.
E_i	İş parçası i 'nin enerji tüketimi.
Ts_i	İş parçası i 'nin başlangıç zamanı.
Te_i	İş parçası i 'nin bitiş zamanı.
$Tend_i$	İş parçası i 'nin bitiş zamanı artı denetleyicinin süresi.
Ta_i	İş parçası i 'nin başlayabileceği en erken zaman.
$c_{i,k}$	Eğer iş parçası i iş parçası k 'dan sonra biterse 1; değilse 0.
r_i	Eğer iş parçası i kopyalandıysa 1; değilse 0.
ch_i	Eğer $c_{i,k}$ ve r_i 1 ise 1; değilse 0.
$a_{i,k}$	Eğer iş parçaları i ve k aynı işlemciye eşlendiyse 1; değilse 0.
TE	Toplam enerji
TR	Toplam güvenilirlik (Kopyalanan iş parçası sayısı)
U	Toplam işlemci sayısı.
u_j	Eğer bir iş parçası işlemci j 'ye eşlendiyse 1; değilse 0.
$Umax$	Kullanılabilecek maksimum işlemci sayısı.

Aşağıdaki (5) ve (6) numaralı ifadeler bu kısıtlamaları kapsar. (7) ve (8) numaralı ifadeler ise sırasıyla iş parçasının çalışma süresini (EUÇS) ve enerji tüketimini gösterir.

$$\sum_{j=1}^m \sum_{v=1}^o b_{i,j,v} = 1, \quad \forall i: 0 \leq i < n \quad (5)$$

$$\sum_{j=1}^m \sum_{v=1}^o b_{i,j,v} \leq 1, \quad \forall i: n \leq i < 2n \quad (6)$$

$$d_i = \sum_{j=1}^m \sum_{v=1}^o t_{i,j,v} \cdot b_{i,j,v}, \quad \forall i: 0 \leq i < 2n \quad (7)$$

$$E_i = \sum_{j=1}^m \sum_{v=1}^o E_{i,j,v} \cdot b_{i,j,v}, \quad \forall i: 0 \leq i < 2n \quad (8)$$

İş parçalarının çalışmaları için belli bir aralık vardır. Bu aralık iş parçasının başlayabileceği en erken zaman Ta_i ve bitebileceği en geç zaman Tdi ile sınırlıdır. Bu kısıtlamalar (9), (10) ve (11) numaralı ifadelerle sağlanır.

$$Ta_i \leq Ts_i, \quad \forall i: 0 \leq i < 2n \quad (9)$$

$$Te_i = Ts_i + d_i, \quad \forall i: 0 \leq i < 2n \quad (10)$$

$$Te_i \leq Td_i, \quad \forall i: 0 \leq i < 2n \quad (11)$$

Daha önce belirtildiği gibi, bir işlemcinin kendisinin hatalı olma ihtimaline karşı iş parçasının kendisi ve kopyasının ayrı işlemcilere eşlenmesi gerekmektedir. Bunu sağlamak için aşağıdaki ifade kullanılmaktadır:

$$\forall i, j: \sum_{v=1}^o (b_{i,j,v} + b_{n+i,j,v}) = 1 \quad (12)$$

İki iş parçası aynı işlemciye eşlenmiş ise, bu iş parçalarının çalışma zamanları çakışmaz. Yani, bir işlemcide belli bir zamanda sadece bir iş parçası çalışabilir. Bunun için öncelikle iki iş parçasının aynı işlemci üzerine eşlenip eşlenmediğinin bilinmesi gerekmektedir. Bunun için bir ikili değişken, $a_{i,k}$, tanımlanmıştır. Eğer iş parçaları i ile k aynı işlemciye eşlenmişse $a_{i,k}$ bir olacaktır. (13) numaralı ifade bunu sağlar. Aynı işlemciye eşlenen iş parçası k 'nin iş parçası i çalışmasını bitirdikten sonra başlaması gerekmektedir. Bir iş parçasının denetleyici süresi de göz önüne alınarak bitiş süresini $Tend_i$ ifade etmektedir. (14) numaralı ifade bu zamanlama kısıtlamasını sağlamaktadır. Bu ifadeye eğer $a_{i,k}$ bir ise k 'nin başlangıç zamanı i 'nin bitiş zamanından büyük demektir. Eğer sıfırsa, zaten bu iki iş parçası aynı işlemci üzerinde değildir ve bitiş zamanları sistemin bitiş zamanından, $Pmax$, küçük olmalıdır.

$$a_{i,k} \geq \sum_{v=1}^o b_{i,j,v} + \sum_{v=1}^o b_{k,j,v} - 1$$

$$\forall i, k, j: 0 \leq i, k < 2n; 1 \leq j < m; i \neq k \wedge i < k \quad (13)$$

$$Ts_k + Pmax - Pmax * a_{i,k} \geq Tend_i$$

$$\forall i: 0 \leq i < 2n; \forall i: 0 \leq k < 2n; i \neq k \wedge i < k \quad (14)$$

Her bir iş parçası ve kopyasından sonra sonuçların doğruluğunu denetlemek (sonuçları karşılaştırmak) için denetleyici devreleri eklenmelidir. Bu çalışmada denetleyici devrelerinin süreleri bir birimlik gecikme olarak alınmıştır. Denetleyici devrenin süresini ekleyebilmek için iş parçası ve kopyasının hangisinin daha sonra çalışmasını bitirdiğini bilmek gerekmektedir. Bunun için $c_{i,k}$ ikili değişkeni tanımlanmış olup, iş parçası i iş parçası k 'den sonra çalışmasını bitiriyorsa bu değişken bir olmaktadır. Bu özellik (15) ve (16) numaralı ifadelerde sağlanmaktadır. Daha önce belirtildiği gibi, bazı iş parçaları performans gibi kısıtlamalardan ötürü kopyalanamayabilir. Bunu (17) numaralı ifadedeki r_i değişkeni sağlamaktadır. Eğer r_i bir ise bu iş parçasının kopyası mevcuttur, değilse bu iş parçası kopyalanmamıştır.

$$Te_i - Te_k \leq (Pmax + 1) \cdot c_{i,k} - 1 \quad \forall i: 0 \leq i < n \quad (15)$$

$$Te_k - Te_i \leq P \max - (P \max + 1) \cdot c_{i,k}$$

$$\forall i: 0 \leq i < n \quad (16)$$

$$r_i = \sum_{j=1}^m \sum_{v=1}^o b_{i,j,v} \quad \forall i: n \leq i < 2n \quad (17)$$

Eğer bir iş parçacığı kopyalanmışsa denetleyici devrenin süresi eklenecektir. Bunu (18), (19) ve (20) numaralı ifadeler göstermektedir. Eğer iş parçacığı i , kopyasından önce çalışmasını tamamlıyorsa, değişken ch_i bir olacaktır. (19) ve (20) numaralı ifadelerde denetleyici gecikmesi t_{ch} iş parçacığından sonra veya kopyasından sonra eklenmektedir. Bunun sonucunda bir iş parçacığının bitiş süresi, kopyası ve denetleyici süresi de göz önüne alınarak $Tend_i$ 'dir.

$$ch_i \geq c_{i,k} + r_i - 1 \quad \forall i: 0 \leq i < n \quad (18)$$

$$Tend_i = Te_i + t_{ch} \cdot ch_i \quad \forall i: 0 \leq i < n \quad (19)$$

$$Tend_i = Te_i + t_{ch} \cdot (1 - c_{i-n,i}) \quad \forall i: n \leq i < 2n \quad (20)$$

3.2. Amaç Fonksiyonları

Bu çalışmada üç farklı amaç fonksiyon üzerine durulmuştur. Gerçek zamanlı uygulamalar hedeflendiği için performans metriği her üç fonksiyonun kısıtlamasıdır. Bu amaç fonksiyonlar şöyle sıralanabilir:

1. *Enerji ve güvenilirlik kısıtlamaları altında işlemci sayısını (maliyet) minimize etmek:* Öncelikle tüm iş parçacıkları ve kopyalarının enerji tüketimi toplanır. Güvenilirlik için kopyalanan iş parçacığı sayısı bulunur. (21) numaralı ifade toplam enerjinin, TE , verilen enerji limitinden, E_{max} , düşük olması gerektiğini vurgular. Benzer şekilde, (22) numaralı ifade toplam kopyalanan iş parçacığı sayısının verilen güvenilirlik kısıtlaması R_{min} 'den büyük olması gerektiğini göstermektedir. (23) numaralı ifadedeki u_j ikili değişkeni bir işlemci üzerine herhangi bir iş parçacığının eşlenip eşlenmediğini gösterir. Bu değişkeni kullanarak, (24) numaralı ifadede görülebileceği gibi, toplam kullanılan işlemci sayısını minimize edilebilir.

$$E \max \geq TE = \sum_{i=0}^{2n-1} E_i \quad (21)$$

$$\frac{\sum_{j=1}^m \sum_{v=1}^o b_{i,j,v}}{n} \geq R \min \quad (22)$$

$$\forall i, j: u_j \leq \sum_{v=1}^o b_{i,j,v} \quad (23)$$

$$MIN: U = \sum_{j=1}^m u_j \quad (24)$$

2. *Güvenilirlik ve maliyet kısıtları altında enerji tüketimini minimize etmek:* (25) numaralı ifade verilen kısıtlamalarla bu amaç fonksiyonu sağlar.

$$MIN: TE, TR \geq R \min \text{ ve } U \leq U \max \text{ iken} \quad (25)$$

3. *Enerji ve maliyet kısıtlamaları altında güvenilirliği maksimize etmek:* Aşağıdaki ifade bu amacı karşılamaktadır.

$$MAX: TR, TE \leq E \max \text{ ve } U \leq U \max \text{ iken} \quad (26)$$

Bunlara ilaveten, tüm iş parçacıkları verilen zaman diliminde çalışmalarını tamamlamalıdır (27).

$$\forall i: Tend_i \leq P \max \quad (27)$$

4. Deneysel Sonuçlar

Bu bölümde, oluşturulan sistemi test etmek için yapılan deneyler sunulmaktadır. Deneylerde, sekiz adet işlemci üzerinde sekiz farklı iş parçacığının enerji tüketimi ve EUÇS değerleri alınarak bir deney ortamı oluşturulmuştur. Bu sekiz iş parçacığının sekiz farklı işlemci üzerindeki enerji tüketimi ve EUÇS değerleri farklıdır. Enerji ve EUÇS değerleri sistemimizde kullanılmakta olan 3,3V değeri üzerinden alınmış olup, diğer değerler bu gerilim seviyesi temel alınarak hesaplanmaktadır. Deneylerimizde değişik gerilim seviyelerinin sonuç eşlemeye etkisi de incelendiğinden, beş farklı gerilim seviyesi kullanılmıştır. Bunlar sırasıyla, 2,0V, 2,4V, 2,7V, 3,0V ve 3,3V'dur. Değişik gerilim seviyeleri kullanıldığında birçok alternatif üretilebilir. Mesela, $o=2$ iken, 2,0V ve 3,3V kullanılmaktadır. Fakat beş gerilim değeri üzerinden farklı ikili kombinasyonlar oluşturulduğunda on çeşit ihtimal çıkmaktadır. Bu konu üzerinde de çalışmalar yapılmış olup [7], bu çalışmada verilen gerilim sayısına göre gerilim değerleri en mantıklı kombinasyon kullanılmak üzere seçilmiştir. Deneylerde kullanılan gerilim seviyeleri Tablo 2'de verilmektedir.

Tablo 2: Deneylerde kullanılan gerilim değerleri.

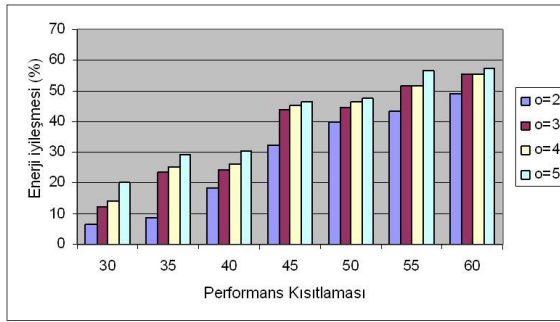
Gerilim seviyesi sayısı	Kullanılan gerilim değerleri
o=1	3.3V
o=2	2.0V, 3.3V
o=3	2.0V, 2.7V, 3.3V
o=4	2.0V, 2.4V, 2.7V, 3.3V
o=5	2.0V, 2.4V, 2.7V, 3.0V, 3.3V

İlk deneyde, enerji ve performans kısıtlamalarında maliyetin değişimi (işlemci sayısı) gösterilmektedir. Bu deneyde $o=2$ ve $o=3$ adet gerilim seviyeleri kullanılmıştır. Tablo 3'de enerji ve performans kısıtlamaları değiştirildiğinde işlemci sayısının nasıl değiştiği görülmektedir. Hücrelerdeki ilk değer $o=2$ iken ikinci değer ise $o=3$ iken kullanılan işlemci sayısını vermektedir. Bu deneyde tüm iş parçacıkları kopyalanmış, maksimum güvenilirlik sağlanmıştır.

Tablo 3: Performans (P_{max}) ve enerji (E_{max}) kısıtlamaları altında işlemci sayıları.

P_{max}	E_{max}					
	50	60	70	80	90	100
50	5;4	4;4	4;4	4;4	4;4	3;3
60	4;4	4;3	4;3	3;3	3;3	3;3
70	3;3	3;3	3;3	3;3	3;3	3;3
80	3;3	3;3	3;3	3;2	3;2	2;2
90	3;3	3;3	3;2	2;2	2;2	2;2
100	3;3	3;2	2;2	2;2	2;2	2;2

Bir sonraki deneyde %100 güvenilirlik kısıtlaması altında üç işlemci kullanarak enerji tüketiminin değişik gerilim seviyelerindeki kazancı test edilmektedir. Şekil 1'de görüldüğü gibi küçük performans azalması sonucu %60'a varan enerji kazancı olmaktadır.



Şekil 1: Performans kısıtlamasında enerji kazancı.

Son deneyde ise enerji ve performans kısıtlamalarının güvenilirlik üzerine etkisi incelenmiştir. Üç işlemci üzerine tüm sekiz iş parçacığı eşlenmeye çalışılmış ve verilen kısıtlamalarda kaç adet iş parçacığının kopyalandığı Tablo 4'de verilmiştir. Bu deneyde iki gerilim seviyesi (2.0V ve 3.3V) kullanılmıştır.

Tablo 4: Performans (P_{max}) ve enerji (E_{max}) kısıtlamaları altında kopyalan iş parçacığı sayısı.

P_{max}	E_{max}				
	60	70	80	90	100
60	1	1	2	2	2
70	2	2	2	2	3
80	3	3	3	3	3
90	3	4	4	4	4
100	4	5	5	5	6

5. Sonuçlar

Bu makalede, heterojen işlemci platformu üzerine bağımsız periyodik iş parçacıklarının eşlenmesi için bir tamsayı doğrusal programlama (TDP) yöntemi sunulmuştur. Deneylerden de görüleceği üzere, sunulan model değişik kısıtlamalar altında değişik amaç fonksiyonlar için kullanılabilir. Sonuçlar, kısıtlamalardaki küçük gevşetmelerin amaç fonksiyonda çok büyük kazançlar sağladığını göstermektedir.

TDP kullanan yöntemlerin genel dezavantajı işlem zamanlarının uzun sürmesidir. Yapılan deneylerde işlem sonuçları iki dakika ile 378 dakika arasında değişmektedir. Çalışılan bilgisayar Unix tabanlı 350Mhz'de çalışan sparcv9 makinesidir. Daha hızlı bir bilgisayar ve daha güçlü bir TDP programcısı bu süreleri azaltacaktır.

6. Kaynaklar

- [1] J.M. Paul, D.E. Thomas, and A. Bobrek, "Benchmark-Based Design Strategies for Single Chip Heterogeneous Multiprocessors", *CODES+ISSS'04*, 2004.
- [2] J. Axelsson, "Analysis and Synthesis of Heterogeneous Real-time Systems", Ph.D. Dissertation, Linköping University, Sweden, 1997.
- [3] M. Berkelaar, K. Eikland, and P. Notebaert, "lp_solve: Open source (Mixed-Integer) Linear Programming system", Version 5.0.0.0. dated 1 May 2004.
- [4] M. Weiser, B. Welch, A. Demers, and S. Shenker, "Scheduling for reduced CPU energy", In *Proceedings of the 1st Symposium on Operating Systems Design and Implementation*, 1994.
- [5] E. Chan, K. Govil, and H. Wasserman, "Comparing algorithms for dynamic speed-setting of a low-power CPU.", In *Proceedings of the First ACM International Conference on Mobile Computing and Networking*, 1995.
- [6] Y. Zang, X. Hu, and D. Chen. "Task scheduling and voltage selection for energy minimization", *Proceedings of 39th DAC*, 2002.
- [7] T. Ishihara and H. Yasuura, "Voltage scheduling problem for dynamically variable voltage processors", *Proc. of the 1998 International Symposium on Low Power Electronics and Design*, 1998.
- [8] E. N. Elnozahy, R. G. Melhem, D. Mossé, "Energy-Efficient Duplex and TMR Real-Time Systems", *IEEE Real-Time Systems Symposium*, 2002.
- [9] C. M. Krishna and Y. H. Lee, "Voltage-Clock-Scaling Adaptive Scheduling Techniques for Low Power in Hard Real-Time Systems", *IEEE Trans. Computers*, 52(12): 1586-1593, 2003.
- [10] K. Dhiraj, N. Pradhan and H. Vaidya "Roll-Forward and Rollback Recovery: Performance-Reliability Trade-Off", *International Symposium on Fault Tolerant Computing*, 1994.
- [11] J. W. S. Liu, *Real-time Systems*, Prentice Hall, 2000.