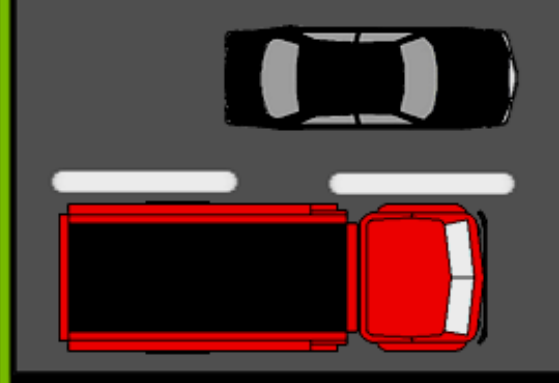


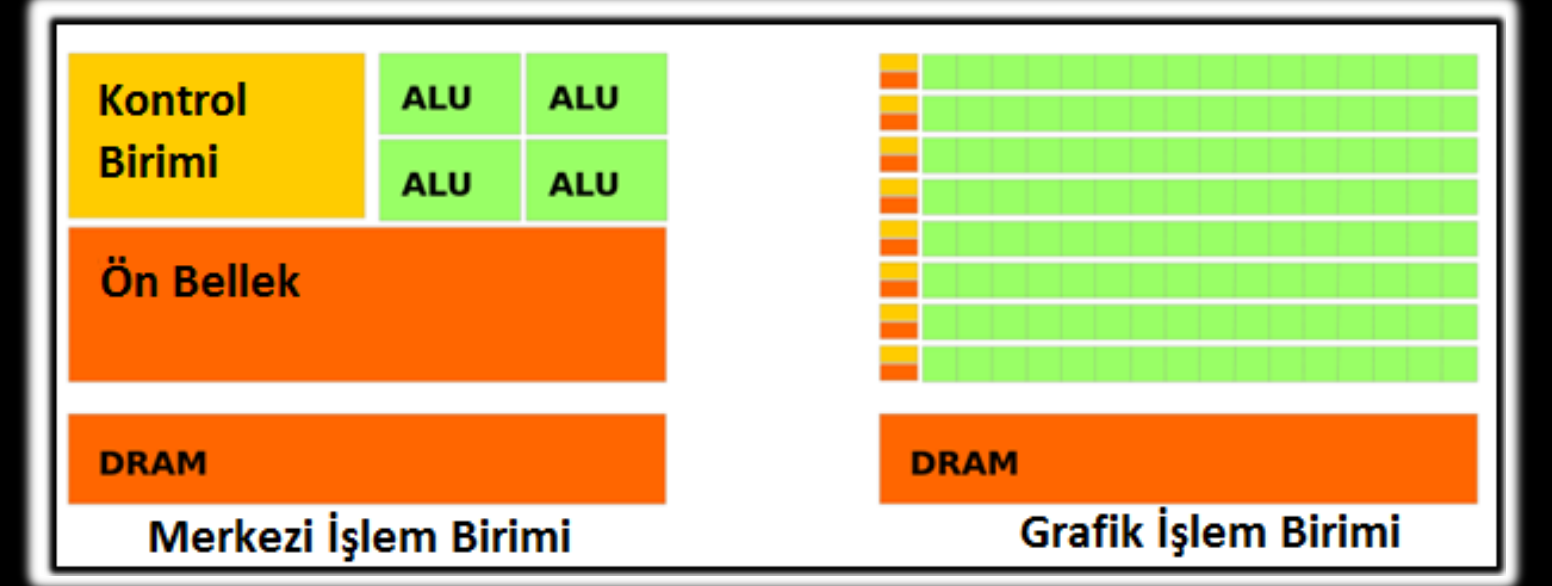
1.ÖZET

Bu projede, konum bilgileri ve kategorileri bilinen araçların yer aldığı bir trafik akışının bilgisayar grafikleri kullanılarak MİB ve GİB üzerinde 2B benzetimi gerçekleştirilmiş, performans karşılaştırılması yapılmıştır. Hücresel otomatlardan faydalanılmış ve NaSch Modeli temel alınarak kodlamalar yapılmıştır. Her bir hücre ya dolu ya da boş olacak şekilde tasarlanmıştır. Hücre durumunun “false” olması hücrenin boş, “true” olması hücrenin dolu olduğunu ifade etmektedir. Şeritler kapalı bir sistem modelinde tasarlanmıştır, bir döngüye sahiptir. Bu modele göre aracın haritadan çıkış yapması durumunda tekrar ilk başladığı noktaya geri dönmesi sağlanmaktadır. Bunun sebebi hücresel otomatın paralel bir güncellemeye ihtiyaç duyuyor olmasıdır. Benzetimde uygulanan hücresel otomat modelinin hücre uzunlukları 7.5 metredir ve güncellenme süresi 1 saniyedir. Modellenen karayolu, Tekirdağ ili Marmara Ereğlisi ilçesi temel alınarak, İstanbul-Tekirdağ karayolunun D110 ile E-84 kodlu yolların kesiştiği bölgedir.

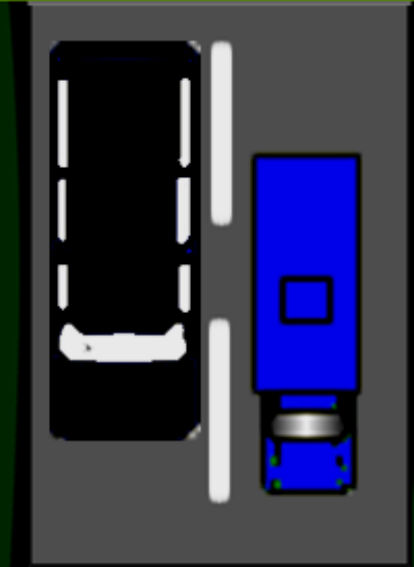


2.GİRİŞ

Projenin öncelikli amacı, otonom bir şekilde ve kurallara uygun işleyen karayolu trafik akışının modellenmesi, bu modeli MİB ve GİB üzerinde uygulayarak performans karşılaştırılmasının yapılmasıdır. Şekil 2-1’de görüldüğü gibi MİB donanımsal olarak seri, GİB ise donanımsal olarak paraleldir. Bu donanımsal yapının gücünden de yararlanılarak CUDA mimarisi ile heterojen hesaplama gerçekleştirilmiştir.

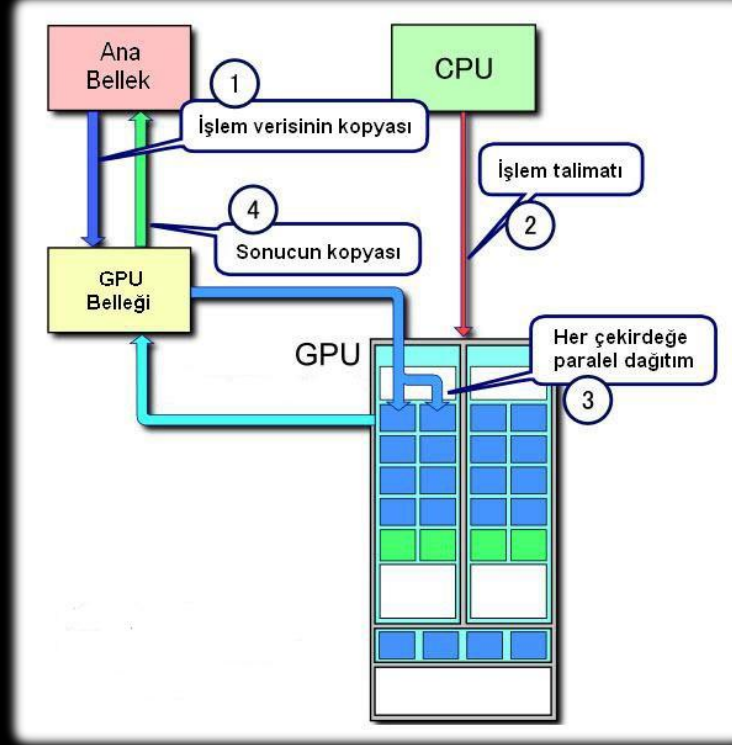


Şekil 2-1: MİB ve GİB’nin donanımsal yapıları



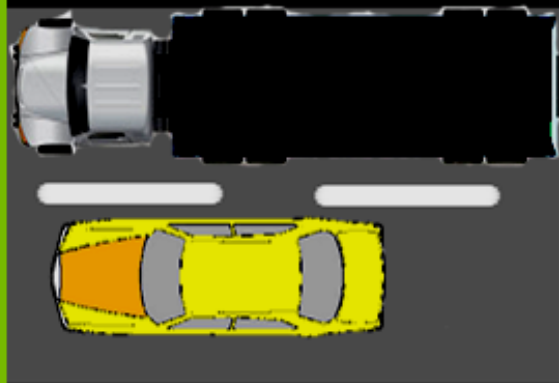
4.CUDA

CUDA(Compute Unified Device Architecture), NVIDIA firmasının 2006 yılında GİB’in donanımsal hesaplama gücünden faydalanmak amacıyla sunduğu heterojen hesaplama mimarisidir. CUDA mimarisinde geliştirilen uygulamalar sadece GİB üzerinde çalışmazlar. Öncelikle MİB tarafından kontrol edilen ana bellek üzerinden grafik kartı üzerindeki belleğe kopyalanması gereklidir.



Şekil 4-1: CUDA işlem akışı

GİB belleğindeki veri CUDA iş parçacıkları tarafından yürütülerek paralel olarak hesaplanması tamamlanır ve ardından tekrar ana belleğe gönderilerek işlem sonlandırılır. Bu akış modeli Şekil4-1’de görülmektedir.



3.HÜCRESEL OTOMATLAR

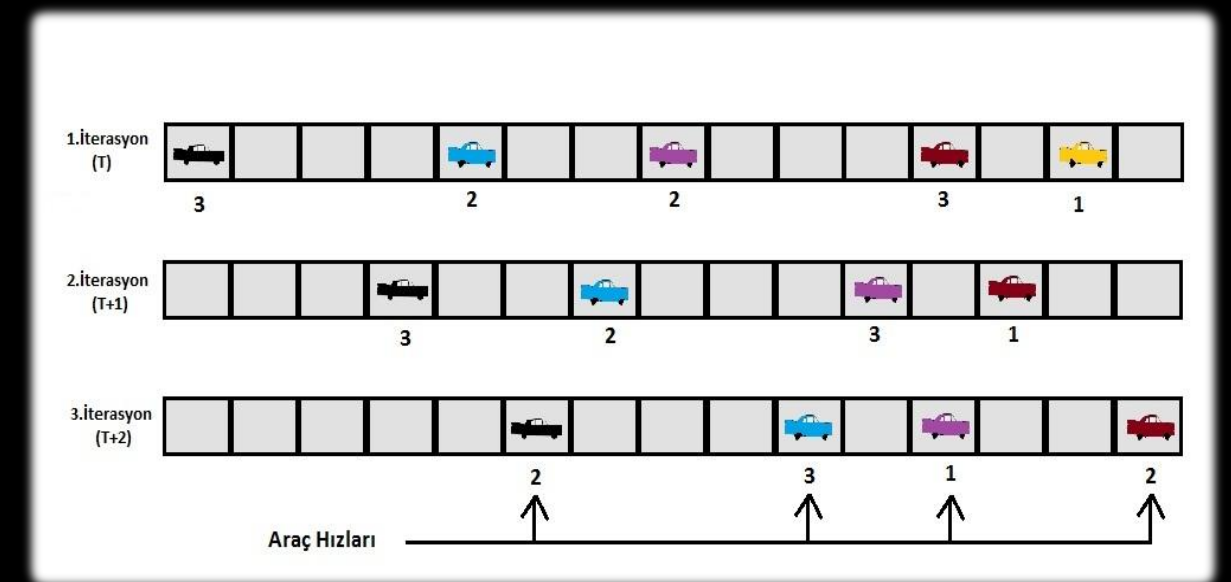
Nagel-Schreckenberg Modeli

Hızlanma: Eğer aracımızın hızı v , v_{max} ’tan düşük ise ve eğer öndeki aracın hızı $v+1$ ’den büyükse, araç hızı bir kademe ilerletilir. [$v \rightarrow v+1$].

Diğer Araçlara Bağlı Olarak Yavaşlama: Eğer i konumundaki bir araç, önünde $j+i$ konumundaki bir aracı görürse($j \leq v$), hızı ($j-1$) kadar azaltılır.

Rasgele Dağılım: p olasılığı ile her bir aracın hızı (eğer sıfırdan büyükse) 1 azaltılır [$v \rightarrow v-1$].

Araç Hareketi: Her araç, v hızını kullanarak konumlarını ilerletirler.

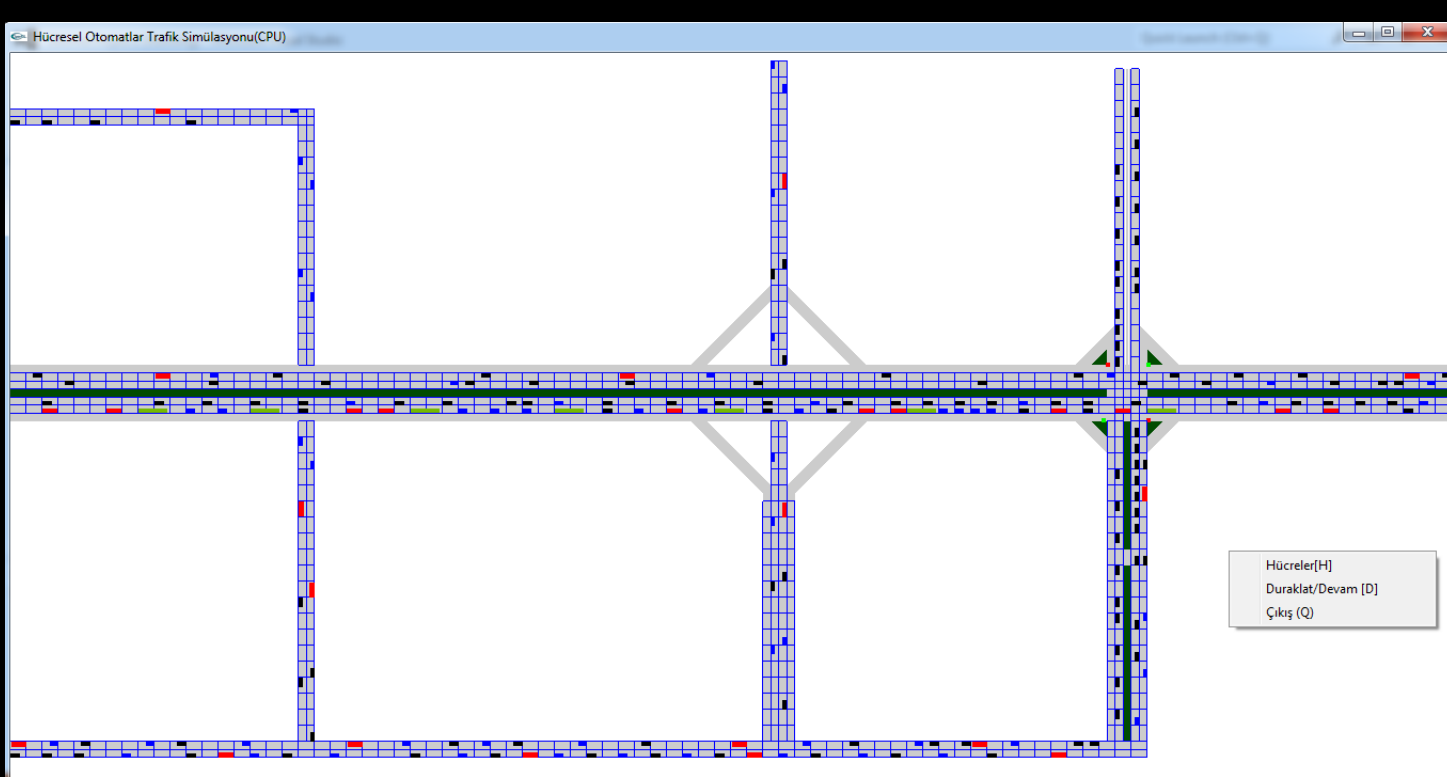


Şekil 3-1: NaSch Modeline Göre Araçların Hız ve Konum Değişimleri

5.BENZETİM

Benzetimde 4 adet araç tipi yer almaktadır. Bunlar; otomobil, tır, kamyon ve kamyonettir.

Kavşaklardan birinin sinyalizasyon sistemi mevcuttur. Şeritlerinde ilerlemekte olan araçlar dönüş yolu gördüklerinde rasgele olarak yön seçmekte ve yönlerine göre ilerlemektedir.



Şekil 5-1: Uygulama Ekran Görüntüsü



6.SONUÇLAR

MİB ve GİB üzerinde 200 adet aracı eş zamanlı güncelleyen fonksiyonun 5’er dakikalık performansı izlenmiş ve bulgular aşağıdaki grafikte ifade edilmiştir. Sonuç olarak MİB ile karşılaştırıldığında GİB üzerinde CUDA mimarisi sayesinde yaklaşık olarak 17 kat hızlanma elde edilmiştir.

