

Evaluation of Metro Lines with Swarm Intelligence Approach

Sena Kaynarkaya¹, Gülen Çağdaş²

0000-0001-2345-6789¹, 0000-0001-8853-4207²

¹ Istanbul Technical University, Graduate School of Science, Engineering, and Technology, Department of Informatics, Architectural Design Computing, Istanbul, Turkey

² Istanbul Technical University, Graduate School of Science, Engineering, and Technology, Department of Informatics, Architectural Design Computing, Istanbul, Turkey

Computer technologies have an important place in the fields of architecture and urban design as in every field today. It is possible to see that they are largely inspired by the life cycles of living organisms, when we look closely at the development processes of these technologies in the fields of architecture and urban design.

Slime mould which are examples of emerging living systems; shows an example of swarm intelligence, which arises from basic needs and produces results very different from individual behavior. Rail systems that significantly reduce the burden of today's urban transportation networks, are constantly changing and renewed, especially in large cities. When we look at the examples of which slime mould are used, it is seen that they have a very important role in the design of metro lines and highway systems.

In the article, it is aimed to evaluate an existing metro line by modeling slime mould behaviors with swarm intelligence approach in digital environment. Based on the studies conducted to evaluate the conformity of the metro lines and highways design criteria, this study will work on the Göztepe-Ümraniye metro line in Istanbul, which is still under construction. The stations on the line were kept constant and the model was expected to create a route, and the compliance of the route with the metro design criteria was evaluated. At points that do not comply with the design criteria, commands are introduced on the model via Python code, and the similarities and differences between the route created by the swarm and the existing route are compared.

Keywords: Swarm Intelligence, Slime Mould, Railway Design.

Received: 24.08.2020

Accepted: 30.09.2020

Corresponding Author:

senakurkcuoglu@hotmail.com

Kaynarkaya, S. & Çağdaş, G. (2020). Evaluation of Metro Lines with Swarm Intelligence Approach. JCoDe: Journal of Computational Design, 1(3), 17-48.

Sürü Zekâsı Yaklaşımı İle Metro Hatlarının Değerlendirilmesi

Sena Kaynarkaya¹, Gülen Çağdaş²

0000-0001-2345-6789¹, 0000-0001-8853-4207²

¹ İstanbul Teknik Üniversitesi, Fen Bilimleri Enstitüsü, Bilişim Anabilim Dalı, Mimari Tasarımda Bilişim, İstanbul, Türkiye

² İstanbul Teknik Üniversitesi, Fen Bilimleri Enstitüsü, Bilişim Anabilim Dalı, Mimari Tasarımda Bilişim, İstanbul, Türkiye

Bilgisayar teknolojileri günümüzde her alanda olduğu gibi mimari ve kentsel tasarım alanlarında da önemli yer tutmaktadır. Bu teknolojilerin mimari ve kentsel tasarım alanlarında gelişim süreçlerine daha yakından bakıldığında büyük ölçüde canlı organizmaların yaşam döngülerinden esinlenildiğini görmek mümkündür.

Beliren canlı sistem örneklerinden olan cıvık mantarlar; temel ihtiyaçlarından doğan ve bireysel davranışlarından çok farklı sonuçlar doğuran sürü zekası örneği sergilemektedir. Günümüz kent içi ulaşım ağlarının yükünü önemli ölçüde hafifleten raylı sistemler, özellikle büyük kentlerde sürekli olarak değişmekte ve yenilenmektedir. Cıvık mantarların kullanıldığı örnekler bakıldığında, metro hatları ve otoyol sistemlerinin tasarımında oldukça önemli bir rolü olduğu görülmektedir.

Makalede sürü zekası yaklaşımı ile cıvık mantar davranışları sayısal ortamda modellenerek mevcut bir metro hattının değerlendirilmesi amaçlanmıştır. Metro hatlarının ve otoyolların tasarım kriterlerine uygunluğunu değerlendirmek için yapılan çalışmalardan yola çıkılarak, İstanbul kentinde yapımı devam etmekte olan Göztepe-Ümraniye metro hattı üzerinde çalışılmıştır. Hat üzerindeki istasyonlar sabit tutulup modelin bir güzergâh oluşturması beklenerek ve oluşan güzergâhın metro tasarım kriterlerine uygunluğu değerlendirilmiştir. Tasarım kriterlerine uygunluk göstermeyen noktalarda model üzerinde Python kodu aracılığı ile komutlar tanıtılıp, sürünün oluşturduğu güzergâh ile mevcut güzergâh arasındaki benzerlik ve farklılıklar karşılaştırılmıştır.

Anahtar Kelimeler: Sürü Zekası, Cıvık Mantarlar, Metro Hattı Tasarımı.

Teslim Tarihi: 24.08.2020

Kabul Tarihi: 30.09.2020

Sorumlu Yazar:

senakurkcuoğlu@hotmail.com

Kaynarkaya, S. & Çağdaş, G. (2020). Sürü Zekası Yaklaşımı ile Metro Hatlarının Değerlendirilmesi. JCoDe: Journal of Computational Design, 1(3), 17-48.

1. GİRİŞ

Mimarlık alanında biyolojiden esinlenme her zaman söz konusu olmuştur. Bilgi ve iletişim teknolojilerinin mimari ve kentsel tasarım alanlarında kullandığı yaklaşımlarda ve modellerde canlı organizmaların yaşam döngülerinden ve davranışlarından esinlendiğini görmek mümkündür. Hem biçimsel hem de ilişkisel bazda karşılaştırıldıklarında, mimarlık ve biyolojinin morfolojiyle yakından ilişkili olduğu görülmektedir. Ayrıca her iki disiplin de bütün halinde çalışan bireysel öğelerden var olur ve bu öğeler arasında sürekli bir etkileşim söz konusudur. Tüm bu benzer özelliklerinin yanı sıra biyolojik süreçler daha hareketli ve üretkendir. Gödel'in (1931) Eksiklik Teorisinde kullandığı "Bazen belirli bir disiplindeki bir sorunu çözmek için tamamen farklı bir bölgeye geçmek zorundasınız." ifadesinden yola çıkıldığında, disiplinler arasındaki iş birliği çoğu zaman kaçınılmazdır.

Doğadaki canlıların, özellikle sürü davranışı sergileyen kuşlar, balıklar ve karıncaların davranışlarının bilgisayar modelleriyle temsili, pek çok kentsel ve mimari tasarım probleminin çözümünde önemli bir esin kaynağı olmuştur.

Makalede, sürü zekâsı yaklaşımıyla, bilgisayar ortamında geliştirilen bir modelle mevcut metro hatlarının güzergahının değerlendirilmesi amaçlanmıştır. Bu amaçla, öncelikle literatürde sürü davranışının fiziksel ortamda benzetimini yapan araştırmalar incelenmiş; daha sonra da sürü davranışı ve bu davranış bağlamında yaşam döngüsüne sahip civik mantar davranışları incelenerek bilgisayarda sayısal bir model geliştirilmiştir. Modelin geliştirilmesinde Rhinoceros / Grasshopper ve Python kodu kullanılmıştır. Model mevcut bir metro hattının verileri ile uygulanmış ve sonuçlar kıyaslanarak değerlendirilmiştir.

2. LİTERATÜR

Günümüzde biyolojik organizasyonların yaşam döngüleri birçok alanda araştırmacılara ilham kaynağı olmuştur. Sürü davranışı sergileyen canlıların yaşamlarını sürdürmek adına gerçekleştirdikleri temel ihtiyaca dayalı basit davranışların, karmaşık sonuçlara neden olduğu gözlemlenmektedir. Bu sistemlerin kendini örgütleyerek sürü zekâsı yaklaşımıyla merkezi bir otoriteye sahip olmadan problem çözme davranışında bulunmaları, ulaşım ve iletişim gibi mühendislik hesapları,

finansal yapılanmalar, benzetim modelleri ve yapay zekâ teknolojilerine referans olmuştur.

2.1 Sürü Zekâsı

Sürü zekâsı; çok sayıda etmenin ortaklaşa gerçekleştirdikleri davranışlar sonucu meydana gelen davranıştır. Sürü zekâsı, ilk olarak 1898 yılında Gerardo Beni, Susan Hackwood ve Jing Wang tarafından beliren fenomenleri tanımlamak için oluşturulan robotik sistemlerde kullanılmıştır (Beni, Wang, 1990).

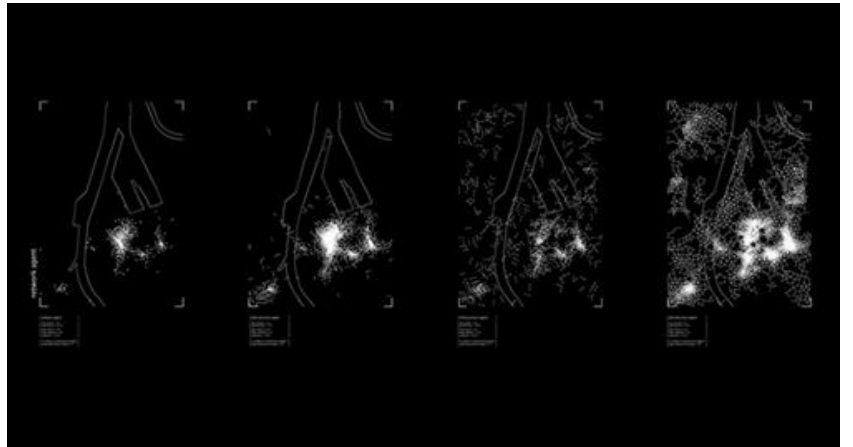
Sürü zekâsı; çevrelerinin farkında, kendini etraftan soyutlamamış, özerk veya yarı özerk olan ve küresel davranışlar ortaya çıkarabilecek etmenlerden oluşur. Bu davranış ile merkezi bir otorite gerekmeden, sadece etmenlerin etkileşimi ile tanımlanan problemin çözümü mümkündür. Sürü zekâsı öz-örgütlenme ve Stimerji kavramları ile ilişkilidir. Stimerji; karınca kolonilerinin ve cıvık mantarların yaşam döngülerinin alt yapısını oluştururken, aynı zamanda sürüdeki bireylerin iletişim kurmalarını sağlayan etkileşimlerin bütünüdür (**Şekil 1**). Stimerji kavramında bilginin dağıtımı esastır. Değişen çevre koşullarından etkilenen etmenin bıraktığı iz sayesinde, tüm sistem bu etkilenmeden haberdar olur ve kendini yeniden organize eder. Karıncaların bıraktıkları feromon, Stimerji örneklerinden birisidir. Öz-örgütlenme, herhangi bir dış kontrol ve yönlendirme olmadan bir sistemin iç düzenine ait belirli bir fonksiyonunun gelişmesi için çevreye adapte olma sürecine denir (Hensel, v.d., 2004). Öz-örgütlü sistemlerde gelişim; rastgelelik, geri bildirim ve etkileşimlerle olur. Dolaylı etkileşim barındıran Stimerji, öz-örgütlenmeyi besleyen bir kavramdır. Sürünün kendini örgütleyerek çalışması ile merkezi otorite ihtiyacı kalmamış olur. Yerel etmenlerin örgütlü etkileşimleri küresel sonuçlar doğururken aynı zamanda sistemi dengede tutar. Sürüyü oluşturan bireylerin her biri özdeştir. Otonom davranışlara sınırlı sahip olmaları ve grup olarak hedefe hizmet edebilmeleri küresel davranışı ortaya çıkarır (**Şekil 1**).

Şekil 1: Öz-Örgütlülük: Kuş sürüleri (Url-1).



Günümüzde belirme teriminin dijital hesaplamalarda ve sosyal davranışları açıklamalardaki kullanımı oldukça hız kazanmıştır. Şehirlerin beliren akıllı sistemler olarak görülmesi ve bir tasarımcıya bağlı kalmadan şekillendiği düşüncesi ile sürü istihbaratının merkezi bir kontrol mekanizması olmadan kendini organize eden yapısının gösterdiği benzerlik, şehircilik için önemli bir referans kaynağı olmuştur.

Sürü zekâsı ve bu istihbarata dayanan algoritmaların şehircilikte kullanımının artması tasarımcı iradesine dayanan planlamanın yerini bireylerin etkileşimine bırakmıştır. Bu bağlamda master plan üretmek yerine master algoritma üretme fikri ile yola çıkılarak, Roland Snooks ve Robert Stuart-Smith tarafından Melbourne’ de tasarlanan Swarm Urbanism projesinin amacı öncelikle ulaşım ve altyapı ağlarını tasarlamaktır. Çalışma ile tasarımı haritalandırmaktan ziyade, etmenlerle elde edilen kolektif tasarım bir sisteme dönüştürülmek istenmektedir (Şekil 2).



Şekil 2: Swarm Urbanism.

Sistemi organize etmek için genel etmen türü kullanılmamıştır. Bunun yerine etmenlerle küresel özelliklerin elde edilebileceği bir ekoloji

oluşturulmuştur. Etmelerin özelliklerinin yanı sıra yoğunluklarının sağladığı hiyerarşik yapı da tasarım sürecinde etkin rol oynamıştır.

Sürü zekasının en kritik noktalarından olan etmen sayısının fazlalığı, kent sistemlerinin barındırdığı birçok unsurun, kentin şekillenmesine olan etkisini açıklamayı kolaylaştırmaktadır.

2.2 Cıvık Mantarların Davranışı ve Uygulama Örnekleri

Doğada, pek çok canlı yaşam faaliyetlerini sürdürebilmek için besin arayışına girerler. Bu arayışı gerçekleştirirken çevrelerine adapte olan ve büyüyen stratejiler geliştirirler. Sahip oldukları basit yaşam döngüleri ile daha küresel sistemler oluşturan ve çevresiyle etkileşim kurarak durumunu değiştiren canlı türlerinden biri de 'Physarum polycephalum' türü olan 'Cıvık Mantarlar'dır. Mantarlar dağınık besin kaynakları arasındaki iletimi yapmak için doğrudan besinler arasında ana bağlantılar ve ek bağlantılar sağlayan ağ hattına sahiptirler.

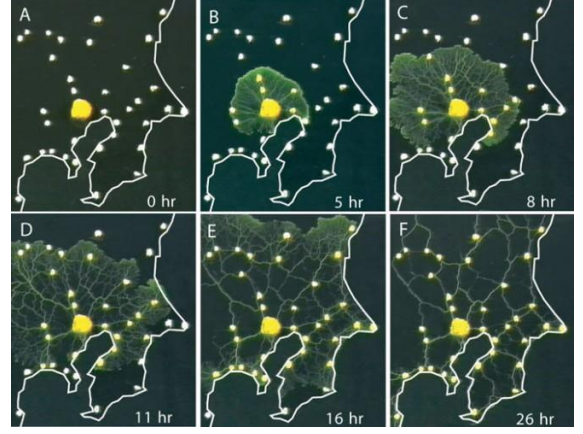
Cıvık mantarlarda besin aktarımı için gelişmiş olan ağ yapısı; taşınan besin miktarı ve taşıma performansı bakımından en verimli olan tüplerden oluşur. Verimliliğin yanı sıra bu bağlantıların en önemli özelliği, besin kaynakları arasındaki en kısa mesafeyi kurmalarıdır.

Tek hücreli canlı organizmalar olan cıvık mantarlar; herhangi bir sinir sistemine sahip değildir. Merkezi bir yönlendirme ve beyin

fonksiyonu barındırmamalarına rağmen, öğrenme ve öğrendikleri bilgileri birleştikleri anda diğer hücrelere aktarma potansiyelleri ile birçok araştırmaya konu olmuşlardır.

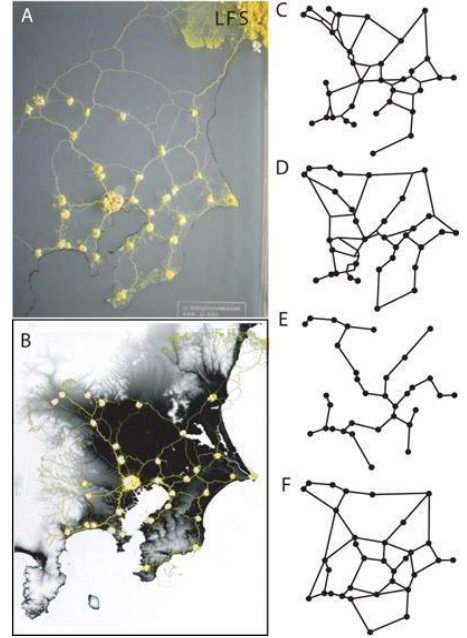
Cıvık mantar hücrelerinin, Nagakaki ve Toth (2000) tarafından yapılan labirent çözme deneyindeki başarısından sonra Tokyo metrosu üzerine bir grup araştırmacı çalışma başlatmıştır (American Association, 2010). Mühendislerin uzun mesailer harcayarak tasarladıkları bu ağ haritalarını cıvık mantar kolonilerinin üretip üretemeyeceği soruları üzerinden gidilerek deneysel birtakım çalışmalar yapılmıştır. Tokyo ve çevresindeki şehirlerin haritası üzerinde cıvık mantar yayılımı gözlemlenmiştir (**Şekil 3**).

Şekil 3: Tokyo metrosu üzerinde zamana bağlı cıvık mantar gelişimi (Url-2).



Cıvık mantarlar tarafından üretilen ağın Tokyo metrosuna oldukça benzer şekilde oluştuğu gözlemlenmiştir. Metro ağı ile mantar ağı arasındaki benzerliğin bu kadar fazla olması, cıvık mantar davranışının matematiksel bir model barındırdığını düşündürmüştür. Tokyo metro hattı üzerinden incelemelere devam edildiğinde, ilk etapta mantar hücrelerinin ağlarını her yere gönderdiği gözlemlenmiştir. Zamanla besin kaynakları arasındaki ağların en sağlam ve kısa olanı korunarak diğer tüpler geri çekilmiştir (**Şekil 4**).

Şekil 4: Zamana bağlı olarak Plazmodyum tüplerinin değişimi (Url-3).



Çalışmalar sonucunda, bu basit organizmaların uzun hesaplamalar ile üretilen bir hattı çok kısa bir sürede oluşturmuş olması, tasarım aşamasının yanında, gelecekte, doğal afet ya da savaş gibi durumlarda

farklı ağ arayışlarında hızlı bir şekilde alternatif geliştirebilecek bir alt yapı sunmaktadır. Ayrıca merkezi olmayan bir irade ile gelişen sistem, yapay zekâ gibi teknolojilere algoritmik bir altlık sağlayabilecektir.

Adamatzky, dünya üzerindeki çeşitli ülkelerin otoyol ağlarının oluşumlarını civık mantarlar üzerinden değerlendiren birçok çalışma yapmıştır. Çalışmalarda belirlenen ülkelerin yoğun nüfuslu, ekonomik potansiyeli yüksek ve ulaşım açısından kritik olan şehirleri seçilerek, civık mantar ağları ile bu ülkelerin otoyol ağları karşılaştırılmıştır. Otoyol ağlarının karşılaştırılması için yapılacak deneye başlamadan önce civık mantarların gelişimi için uygun ortam hazırlanmıştır. 120x120 mm boyutlarında kare ya da 90 mm çapında yuvarlak polistiren petri kaplarına içme suyu serpilmiş kâğıt havlu yerleştirilmiştir. Yulaf gevreği konularak civık mantarlar yetiştirilmek üzere gevreklerin üzerine yerleştirilmiştir. Deneyi yapılacak bölgenin haritası belirli ölçekte küçültülerek agar plakalar halinde kesilmiştir (**Şekil 5**). Hattın geçtiği şehirlerin yerlerine yulaf gevreği yerleştirilmiştir. Deneyin görselleştirildiği zamanlar dışında, deney ortamı karanlık ve sıcaklığı 22-25° C dir. Deney, sonuçlara göre ortalama 20-25 sefer tekrarlanmıştır. Deney kapları tarayıcıda taranıp doygunluk seviyesi yükselttilerek görselleştirilmiştir (Adamatzky, Akl, 2011).



Şekil 5: İngiltere'nin agarlardan kesilmiş haritası (Url-4).

Kesilen agar plakalar üzerinde seçilen başlangıç şehrindeki yulaf üzerine civık mantar aşılansak deney başlatılmıştır. Toplanma eylemini tetikleyen yulafın yanı sıra engel olacak bölgelerden mantarların geçmemesi için, civık mantarları uzaklaştırıcı etkisi bulunan kristal tuz kullanılmıştır. Otoyol hattının mevcut hali H grafiği olarak kabul edilerek deneye başlanmış ve civık mantar grafiği tanımlanırken aşağıdaki kabuller yapılmıştır:

- U: Kentsel alan kümesi.
- E: Set kenarı.
- P: Cıvık mantar grafiği
- $w: E[0,1]$: Kenarların olası ağırlığı. E' nin her bir kenarı bir olasılıkla ilişkilendirilir.
- P(Q): Eşik değeri. U bölgesindeki noktalar arasında oluşan tüplerin kalınlığıdır. Tüp kalınlıkları ve ağırlıkları ile orantılıdır. Q değeri değiştiğinde noktaların bağlanma şekillerinde değişiklik gözlenmektedir.
 - $P(Q) = 0$ iken H grafiği ham haldedir.
- $T(E) = e$ ve $E: w(e) > Q$ iken Q' ya eşit veya daha az ağırlığa sahip olan tüm kenarları kaldırır.

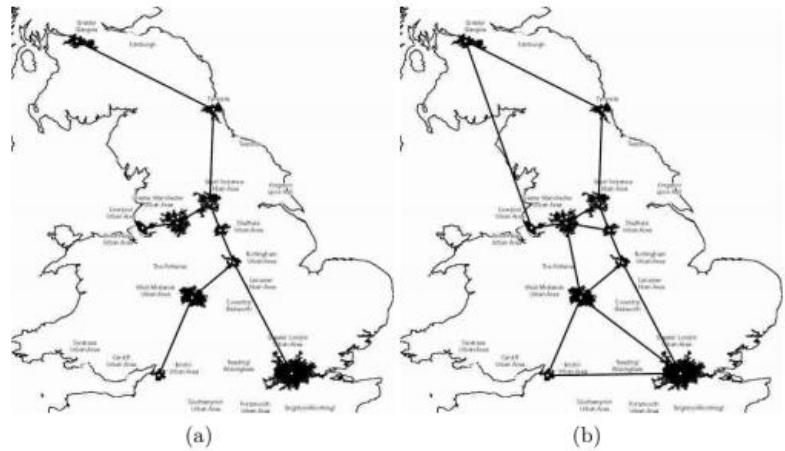
Cıvık mantar grafiği şu şekildedir:

$$P(Q) = (U, T(E), w)$$

Cıvık mantar grafiğinden elde edilen sonuçlar otoyol yapımlarında kullanılan Düzlemsel Yakınlık Grafikleri (Gabriel Grafiği, Göreceli Mahalle Grafiği) ile karşılaştırılarak sonuçlar elde edilir (Adamatzky, Prokopenkob, 2011).

Bu çalışmalar, İngiltere (Adamatzky, Jones, 2009), (**Şekil 6**), Meksika (Adamatzky, A., 2010), Kanada (Adamatzky, Akl, 2011), Avustralya (Adamatzky, Prokopenkob, 2011), Hollanda (Adamatzky, v.d., 2012), Çin (Adamatzky, v.d., 2013), gibi ülkelerin otoyolları için tekrar edilmiş ve büyük oranda benzerlikler olduğu belirtilmiştir.

Şekil 6: (a) Canlı organizmaların dağılımını incelemek için kullanılan İngiltere haritası (b) Gabriel grafiği karşılaştırması (Url-5).



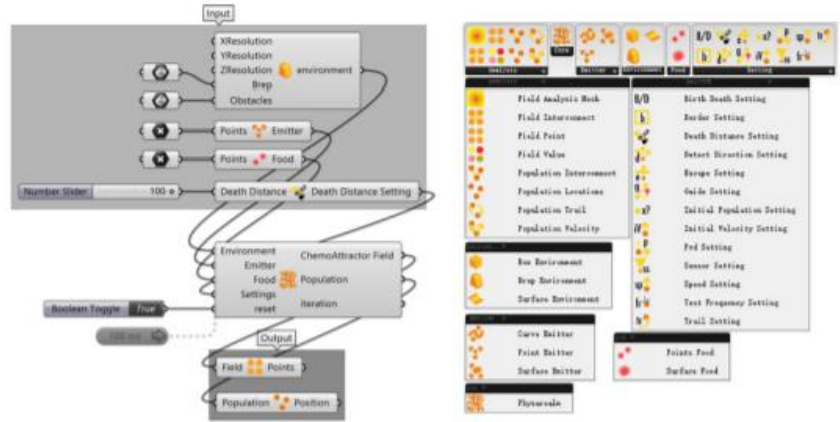
3. METRO HATTI DEĞERLENDİRME YÖNTEMİ

Yürütülen çalışmada sonuç ürüne ulaşırken çeşitli aşamalardan geçilmiştir. Problemin belirlenmesi ve probleme nasıl yaklaşıp çözüme nasıl ulaştırılacağı gibi kararlar süreçle beraber şekillenmiştir. Bu nedenle tezin anlaşılabilirliğini artırabilmek adına Şekil 7 de verilen diyagram oluşturulmuştur. Tez çalışması ve buna bağlı model üretilirken belirme kavramı üzerinden gidilerek, bu kavrama referans veren bir yaşam döngüsüne sahip olan civik mantarların, tez çalışmasındaki benzer birçok çalışmada kullanılması üzerinden gidilmiştir. Alınmak istenen sonuç ürüne verimli şekilde ulaşılacak bir çalışma ortamı belirlenerek eklenti içerisinde bulunan parametreler kontrollü şekilde değiştirilmiştir. Değişimlerden elde edilen geri dönüşler sonucunda parametreler güncellenerek sürekli tekrar eden bir süreç izlenmiştir. Sürünün üremesi için uygun parametreler elde edildikten sonra, popülasyon çoğalırken ortaya çıkan problemler belirlenerek, problemlerin çözümleri üzerine gidilmiştir. Sürünün çoğalırken gösterdiği fazla yayılım ve metro tasarım kriterleri de göz önüne alınarak civik mantar sürüsünün çizdiği metro güzergahı gözden geçirilerek, belirli kısıtlar ve yönlendirmeler ile güzergâh etmenlere tekrar tekrar çizdirilmiştir.

Şekil 7: Çalışmada İzlenen Yöntem.



Etmenlerin aralarındaki uzaklık, konumları arasındaki açılar ve iz miktarının yoğunluğuna göre farklı üreme şekilleri ve doğum oranı göstermeleri öncelikle kullanılacak programda geri bildirim ihtiyacı doğurmaktadır. Bu nedenle çalışmanın modelini oluşturmak için ortam olarak; eş zamanlı çıktı alma ve parametrik modelleme fırsatı sunan Rhinoceros-Grasshopper tercih edilmiştir. Grasshopper, Rhinoceros'a entegre edilmiş bir modelleme aracıdır. David Rutten tarafından geliştirilen Grasshopper, sürekli olarak parametreleri güncelleme ve güncelleme sonucu ortaya çıkan değişiklikleri anında görüntüleme imkânı sunar. Programın verdiği eşzamanlı geri bildirimler sayesinde geometri üzerinde yapılan değişiklikler, kaydırıcılar sayesinde sürekli olarak güncellenmektedir. Oluşturulan algoritma setleri adım adım birbirine bağlanarak modelin çalışması sağlanmaktadır. Modelleme Rhinoceros'ta yapılabileceği gibi, doğrudan Grasshopper üzerinden de yapılabilmektedir. Modelin geliştirilmesinde, Grasshopper'a gömülerek çalışan ve civik mantarlar için üretilmiş olan Physarealm eklentisi kullanılmıştır. Physarealm; etmen tabanlı algoritmalar ile üretilmiş bir eklentidir. Civik mantarların yaşam döngüsünü açıklayan kuralları kullanarak laboratuvar ortamında yapılan deneylerin, dijital ortamda benzetiminin yapılabilmesi için kullanılan bir yazılımdır. Altı bileşen grubundan oluşmaktadır: Analysis, Core, Emitter, Environment, Food, Settings (Şekil 8).

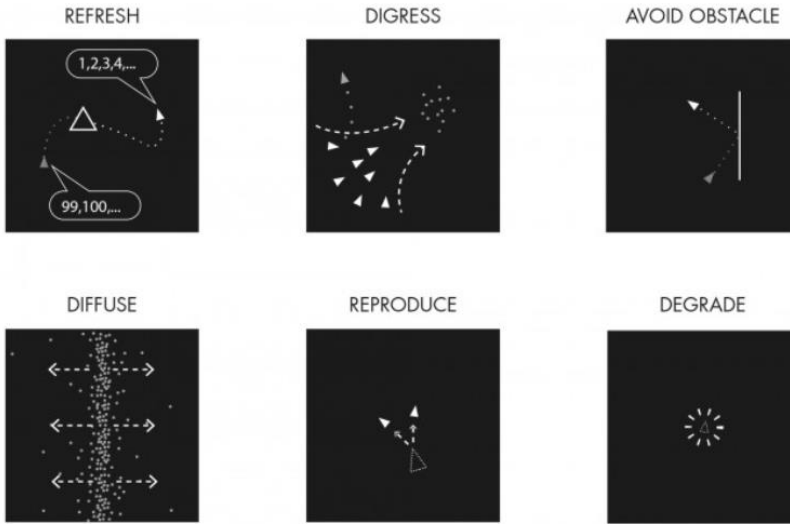


Şekil 8: Physarealm Bileşenleri (Url-6).

Physarealm eklentisini oluşturan bileşenlerden Environment bileşeni; etmenlerin üreyeceği ortamı tanımlarken, Emitter bileşeni; koloninin üremesi için ortama bırakıldığı ilk noktayı ve metro hattının başlangıç istasyonunu temsil etmektedir. Box, brep ya da point ile temsil edilebilen Emitter bileşeni ile mantarların yöneleceği diğer besin kaynaklarını, metro hattının diğer istasyon noktalarını temsil eden Food

bileşeni de Core noduna beraber bağlanarak sistemin çalışmasını sağlamaktadır.

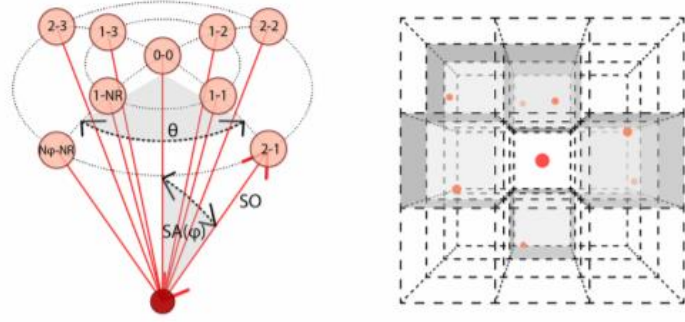
Settings bileşeni; mantar davranışlarını kontrol eden parametreleri ekleyebileceğimiz alt bileşenleri içermektedir. Cıvık mantar davranışlarını formülleştiren Jeff Jones'un (2010) çalışmaları referans alınarak, Setting bileşeninin içindeki parametrelere optimum başlangıç değerleri atanmıştır. Physarealm eklentisi ayarlarının temelini birkaç davranış modeli oluşturmaktadır (**Şekil 9**).



Şekil 9: Physarealm davranış modelleri (Url-7).

Jeff Jones'ın (2010) modelinden esinlenerek üretilse de sonuç olarak iki model birbirinden oldukça farklıdır. Sürüyü oluşturan etmenlerin üreme ortamları, kafeslere bölünmüş bir küptür (**Şekil 10**). Etmenlerin davranış modelleri, bu küpün içerisinde şekillenir. Birbirlerine göre konumları ve bu konumlar arasındaki açılar Physarealm eklentisindeki parametrelerin temelini oluşturmaktadır. Sensör davranışında; etmenler birbirlerinin salgıladığı kemo-atraktif maddeyi algılamalarını sağlayan sensörlere sahiplerdir. Davranış bileşenini dört parametre oluşturur: Etmenin Z ekseninde hareket ettiği bir koordinat düzlemi, her sensöre radyal mesafede olan Algılama Dengesi (SO), maksimum kutup açısında olan Algılama Açısı (SA) ve aynı Z koordinatında bulunan etmenleri algılamayı sağlayan sensör sayısından oluşan Yön Algılama (NPhy)dir.

Şekil 10: Davranışın gerçekleştiği ortam ve davranış açıları (Url-8).


















Doğum ve ölüm davranışında (Adaptasyon); Etmen sayısı çok fazlaştığında ortamdaki etmenlerin bir kısmı ölür. Buna göre algoritma yeni etmen sayısına göre davranışı günceller. Kurallar yakın yarıçap içerisindeki etmenlere göre çalışır. Doğum için yarıçap DvR , ölüm için ise DeR 'dir (**Tablo 3**). Alanın yarıçapı $(2 * DvR + 1)$ ve $(2 * DeR + 1)$ 'dir. Bu nedenle, kontrol alanı, yarıçap=1 ise $3*3*3=27$ 'dir. Komşu etmenlerin sayısı N 'dir. $DvMin > 0$ ve $DvMax < (2 * DvR + 1)^3$ ise büyüme gerçekleşmez. $DeMin > 0$ ve $DeMax < (2 * DeR + 1)^3$ ise etmen sayısı azalır. Örneğin, algoritma 26 komşu hücreyi sayar ve $DeR = 1$ olduğunda başka herhangi bir etmenin bunları işgal edip etmediğini kontrol eder. Komşu hücrelerin sayısı $DvMin$ ve $DvMax$ arasında ise etmen kendini çoğaltır. Kemoatraktan iz davranışında; küpü oluşturan kafesler $(3*3*3)$ ölçülerindedir. $TrRat$ parametresi besin üzerinden geçen etmenin bıraktığı izin miktarıdır. İzin miktarına göre ortaya çıkan davranış ve buna bağlı olarak oluşan kurgu da değişir. Ana davranış parametreleri dışında sürü davranışını değiştiren birkaç parametre daha bulunmaktadır:

Dikey Kılavuz Faktörü (VGF): Etmenin hareketini etkileyen parametredir. Yüksek bir VGF ile etmenler daha dikey olarak hareket etme eğilimindedir; bu da sütunlar gibi formları bulmak için yararlıdır.

Başlangıç Hızı (IV): Bu parametre, akın gibi çok-etmenli sistemlerde etkisi daha fazla olan bir faktördür. Cıvık mantarlarda akın durumu görülmesi de bu davranış yönlü etmenlerin başlatılması için önemlidir.

Kaçış Olasılığı (EscP): Parametre değerinin büyük olması etmenlerin belirlenen sınırlardan kaçmasına veya belirlenen engellere girmesine neden olur.

Fiziksel deney ortamlarında kullanılan materyallere benzerlik ve metro hattının temsiline uygunluğu açısından Physarealm içerisinde üreme ortamı olarak “Surface Environment”, aşılama ve besin kaynaklarını göstermesi için ise “Point Food ve Point Emitter” seçilmiştir. Geri kalan bileşenler üreme kuralları, ortam boyutları ve istasyonlar arası mesafeye uygun olarak belirlenmiştir. Modelin çalışacağı yüzey, etmenlerin hareket edeceği noktalar ve ilişkileri düzenlemek için Grasshopper komutları kullanılmıştır (**Tablo 1**).

	BİLEŞEN	GÖRSEL	İSİM	ID	TANIMLAMA	YER
G R A S S H O P P E R	PARAMS		POINT	Pt	Bu bileşen nokta veya noktaları temsil eder.	Geometry
			SURFACE	Srf	Bu bileşen yüzey veya yüzeyleri temsil eder.	
			PANEL	-	Bileşenleri değerlendirir ve bu değerlendirmeyi bağlanma sırasına göre listeler.	Input
			NUMBER SILIDER	-	Sayı değerini değiştiren kaydırıcıdır.	
			BOOLEAN TOGGLE	Toggle	Tek bir doğru ve yanlış değer arasında hızlı bir şekilde geçiş yapmanızı sağlar.	Primitive
			COLOUR SWATCH	Swatch	Renk kartelası.	
			TIMER	-	Parametreleri bazı aralıklarla değiştirebilir ve sonucu animasyonlu bir film gibi görebilirsiniz.	Util
	MATHS		LARGER THAN	Larger	Büyüktür bileşeni iki veri listesi alır ve A Listesinin ilk öğesinin B Listesinin ilk öğesinden büyük olup olmadığını belirler. İki çıktı, iki listeyi daha büyük bir değere göre değerlendirmek isteyip istemediğinizi belirlemenizi sağlar.	Operators
			SMALLER THAN	Smaller	Küçüktür bileşeni, Büyüktür bileşenin ters hareketini gerçekleştirir. Küçüktür bileşeni A listesinin B listesinden küçük olup olmadığını belirler ve bir boole değerleri listesi döndürür.	
	SETS		CULL PATTERN	Cull	Bir kafesden alınacak yüzler.	Sequence
			GRAFT TREE	Graft	Her öğeye fazladan bir dal ekleyerek bir veri ağacı oluşturur.	Tree
	VECTOR		CLOSEST POINTS	CPs	Bir nokta koleksiyonunda en yakın noktaları bulur.	Point
	SURFACE		SPHERE	Sph	Küresel bir yüzey oluşturun.	Primitive
			BREP CLOSEST POINT	BrepCP	Bir brep üzerindeki en yakın noktayı bulur.	Analysis
DISPLAY		CUSTOM PREVIEW	Preview	Oluşturulan şeklin renklendirilmesini sağlar.	Preview	

Tablo 1: Grasshopper’ da kullanılan bileşenlerin listesi.

3.1 Değerlendirilecek Metro Hattına Ait Veriler

Makale kapsamında yapılacak çalışma için, elde edilebilecek verilerin güncelliği nedeni ile Göztepe-Ümraniye metro hattı pilot hat olarak belirlenmiştir. İstanbul metrosu tasarım kriterleri ve standartları kitapçığından, metro hattı tasarlanırken dikkat edilmesi gereken hususlar ile ilgili veriler elde edilmiştir. İstanbul Metropolitan Alanı Kentsel Ulaşım Ana Planı doğrultusunda; ilk durağı, son durağı ve durak sayısı belirli olan hat bu kriterlere göre şekillenmektedir. İki durak arası mesafenin maksimum uzunluğu, aracın çıkabileceği en yüksek eğim

değeri ve dönebileceği virajın yarıçapı gibi, her hat için uygulanan genel kurallar ve hattın geçtiği çevrenin durumu, istasyon noktalarını belirlerken dikkat edilen unsurlardır.

Metro A.Ş Etüt Projeler Müdürlüğünden alınan iki adet avan proje ve kesin proje arasında yapılan karşılaştırmalar ile istasyon noktalarını belirlerken dikkat edilen kriterlerin öncelik sıralaması yapılmıştır. Göztepe-Ümraniye hattı tasarlanırken belirlenen kriterler şunlardır: Entegrasyon, Yoğunluk (Okul, alışveriş merkezi, hastane), İmar durumu (Tablo 2).

	İstasyon	Kriter
1	Göztepe 60. Yıl Parkı	Caddebostan sahili lastikli araç yolu entegrasyonu
2	Göztepe	Marmaray entegrasyonu
3	Sahrayıcedid	Minibüs caddesi lastikli araç yolu entegrasyonu
4	Yeni Sahra	Kadıköy-Tavşantepe metro entegrasyonu
5	Ataşehir	İki durak arası maksimum mesafe ve imar durumu
6	Finans Merkezi	Cazibe merkezi
7	Soyak Yenişehir	İki durak arası maksimum mesafe ve imar durumu
8	Atakent	İki durak arası maksimum mesafe ve imar durumu
9	Çarşı	Üsküdar-Ümraniye metro entegrasyonu
10	Hastane	Ümraniye E. A. Hastanesi
11	Kazım Karabekir	Son durak

Tablo 2: Göztepe-Ümraniye hattı tasarım kriterleri.

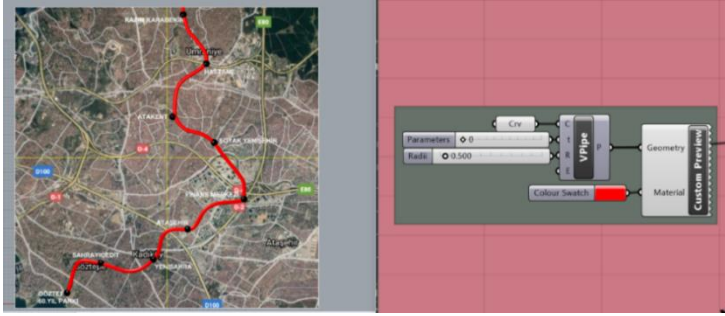
3.2 Metro Hattı Değerlendirme Modeli ve Uygulanması

Bu çalışmada referans canlı olarak belirlenen civık mantarların sürü zekâsı ile ortaya çıkardıkları davranışların esas dayanağı komşuluk ilişkileri ve bu ilişkiden doğan etkileşimlerdir. Civık mantarları temsil eden etmenlerin ortamdaki üreme durumları, ulaşabilecekleri en yüksek sayı ve birbirlerine olan uzaklıkları nedeni ile gerçekleşecek ölümleri, beliren fenomenlerde ortaya çıkan komşuluk ilkesiyle doğrudan ilişkilidir. Makale kapsamında geliştirilen model, metro güzergahları üzerinden yapıldığı için üreme ortamı yüzey olarak seçilmiştir. Eğitim göz ardı edilerek çalışma yapılmıştır. Jeff Jones' un (2010) çalışmasında ve diğer uygulamalarda piksel olarak belirlenen ortam boyutu, bu çalışmalardan farklı olarak santimetre cinsinden

verilmiştir. Parametre ve denklem hesapları piksele dayalı yapıldığı için bazı parametreler bu çalışmaya birebir uymamıştır. Modelin geliştirildiği ortam 90x90cm boyutlarında 1/10.000 ölçeğinde küçültülerek oluşturulmuştur. Metro istasyon yerleri, Metro AŞ.' den alınan veriler doğrultusunda aslına uygun olarak yerleştirilmiştir (Şekil 11). İstasyon ve güzergâh temsilleri şöyledir:

● İstasyon noktaları

■ Güzergâh



Şekil 11: Mevcut hattın Grasshopper' da temsili.

Metro AŞ.' den alınan projenin AutoCad çizimleri Rhinoceros ortamına aktararak istasyon noktaları ve hat görsel olarak daha okunaklı hale getirilmiştir. AutoCad çizimi üzerinden, Grasshopper'da bulunan 'Pipe' bileşeni ile hattı oluşturan eğri, bir tüpe temsil edilmiştir. Yapılan sayısallaştırma üzerinden sürünün oluşturduğu hattın mevcut hatta olan benzerliği ve mevcut hatla arasındaki farklılıklar gözlemlenmiştir. İlk olarak mevcut hat ile Grasshopper'da oluşturulan modeldeki sürünün başlangıç parametreleri dışında herhangi bir müdahale bulunmaksızın oluşturdukları metro hattı karşılaştırılmıştır. Daha sonra belirli noktalara konulan engeller ile sürünün yönelimine bağlı olarak hattın gelişimi gözlemlenmiştir.

Physarealm eklentisinde "Surface Environment" bileşeni için kullanılan başlıca beş ayar bileşeni bulunmaktadır. Her ayar bileşeni kendi içinde üremeyi etkileyen birçok parametre içermektedir. Bu parametreler kemoatraktör izin diğer etmenler tarafından algılanabilmesi ve davranışlarının değişmesi üzerinden oluşturulmuştur (Tablo 3, Tablo 4). Benzetim sürecinin başlangıcında belirlenen etmen sayısı 70'tir. Sürü nüfusunun davranışı etkilemesi nedeniyle parametrelere farklı değerler atanarak elde edilen değerler üzerinden davranışlar gözlemlenmiştir.

Tablo 3: B/D Setting Parametre Tanımları.

B/D Setting

DvR = Doğum mesafesi yarıçapı

DvMin = DvR içinde olan etmenlerin birbiri ile olan en kısa mesafesi

DvMax = DvR içinde olan etmenlerin birbiri ile olan uzun mesafesi

DeR = Ölüm mesafesi yarıçapı

DeMin = DeR içinde olan etmenlerin birbiri ile olan en kısa mesafesi

DeMax = DeR içinde olan etmenlerin birbiri ile olan en uzun mesafesi

Speed Setting

Max Speed = Etmenlerin hareket edeceği en yüksek hız

Death Distance Setting

Death Distance = Pozitif değerdeki ölüm uzaklığıdır

Sensör Setting

Sensor Angle (SA)= Etmenlerin algıladığı maksimum kutup açısıdır.

Rotate Angle (RA) = Etmen en yüksek kemoatraktör miktarının olduğu yere değil dönme açısının olduğu yöne gider.

Sensor Offset (SO)= Sensörün algıladığı aralığın en yüksek değeridir.

Detect Direction Setting

DDirR = Algılanan konik yapının alt çemberinin bölüneceği nokta miktardır.

DDirP = Algılanan konik yapının alt çemberinin bölüneceği açıdır.

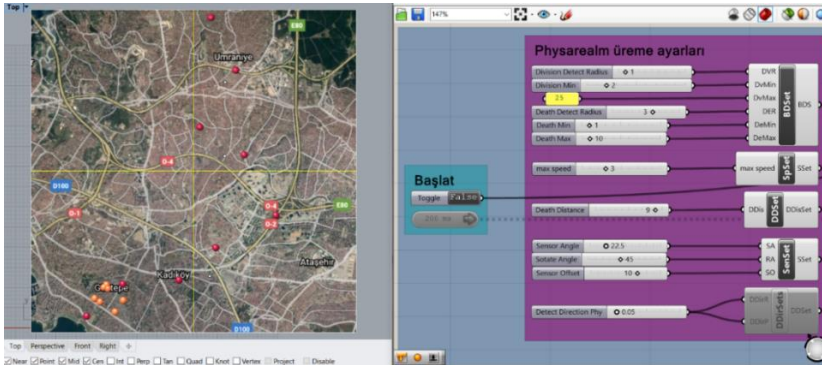
Tablo 4: Speed Setting, DDis. Setting, Sensör Setting ve Detect D. Setting Parametre Tanımları.

Eklentinin geliştiricilerinin verdiği başlangıç verileri üzerinden model geliştirilmeye başlanmıştır. Ortam boyutunun büyüklüğü etmen sayısını ve kullanılan parametreleri oldukça değiştirmektedir. B/D Setting içerisinde bulunan Dvmin ve DvMax parametreleri, ortam boyutuna uygun olarak yapılandırılarak üremenin sağlıklı bir şekilde gerçekleşmesi hedeflenmiştir. Simülasyonun rahatça takip edilebileceği en uygun hız seçilmiştir. Sensor Setting değerleri geliştiricilerin önerdiği değerlere yakın olarak belirlenmiştir (**Tablo 5**).

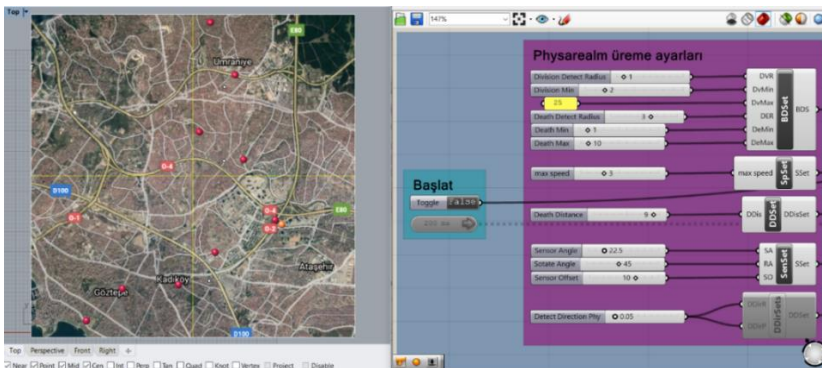
B/D Setting	Speed Setting	Death Distance Setting	Sensor Setting
DvR=1			
DvMin=2			SA=22.5
DvMax=25	Max Speed=3	Death Distance=9	RA=45
DeR=3			SO=10
DeMin=1			
DeMax=10			

Tablo 5: Simülasyon-1' in parametreleri

Değerler tanımlanıp simülasyon başlatıldığında etmenler başlangıç istasyonundan diğer istasyonlara doğru çoğalarak ilerlemeye başlamıştır (**Şekil 12**). Modelin ilk çalışma aşamasında az miktarda çoğalma eğilimi gösterebilir de dördüncü istasyona ulaşmadan sayıları azalmış ve tüm etmenler ölmüştür (**Şekil 13**). DvMax parametresinin yüksek olması etmenlerin uzak mesafelerde de üremesine sürünün bir anda kontrolsüzce büyümesine ve bir anda ölmesine neden olmuştur.



Şekil 12: Simülasyon 1-a



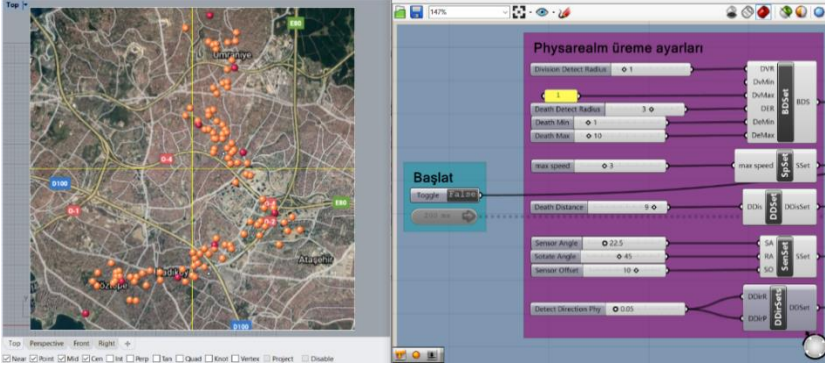
Şekil 13: Simülasyon 1-b

Modelin çalışması sürecinde, parametrelerin bazıları sabit tutularak, bazılarının değerleri değiştirilerek simülasyon sonuçları izlenmiştir; bu süreçte yedi simülasyon gerçekleştirilmiştir. Modelin ana parametreleri olan DvMin, DeMin ve SA parametrelerinin değeri etmenlerin kemoatraktör izi takip edebilmesi açısından oldukça önemlidir. Bu yüzden DvMin, DeMin ve SA parametreleri yedinci simülasyonda değiştirilerek etmenlerin davranışları tekrarlanmıştır (**Tablo 6**). Tanımlanan son parametreler ile etmenlerin düzenli bir şekilde üremeye devam ettikleri ve son istasyona kadar ulaştıkları görülmüştür (**Şekil 14**).

B/D Setting	Speed Setting	Death Distance Setting	Sensör Setting	Detect Direction Setting
DvR=1				
DvMin=0			SA=22.5	
DvMax=1	Max Speed=3	Death Distance=9	RA=45	DDirR=0.05
DeR=3			SO=10	DDirP=0.05
DeMin=1				
DeMax=10				

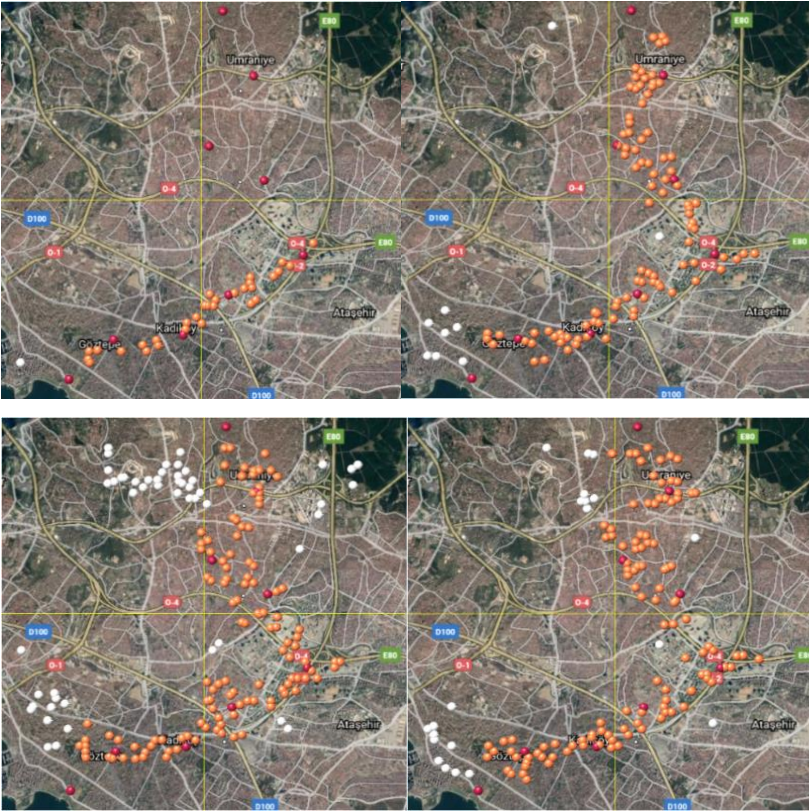
Tablo 6: Simülasyon-7' in parametreleri

Parametreler değiştirildikten sonra, modelin tekrarlı çalıştırılması sonucunda civık mantarların istasyonlar arası düzenli bir ilerleme kaydettiği ilmiştir. B/D Setting parametreleri arasında çok fazla fark olmaması dağılım ve buna bağlı olarak sürünün yok olmaması için önemlidir. Çünkü doğum yarıçapının en az ve en fazla değeri arasındaki fark arttığı zaman etmenlerin birbirinden uzaklaşarak izleri algılayamamasına neden olmuştur. Fakat kemoatraktör miktarının çok yüksek olması etmenlerin hareketi için tek başına en önemli parametre değildir. SA parametresi bu yüzden iz miktarı ile beraber çalışarak etmenlerin üremesine yön verir.



Şekil 14: Simülasyon-7.

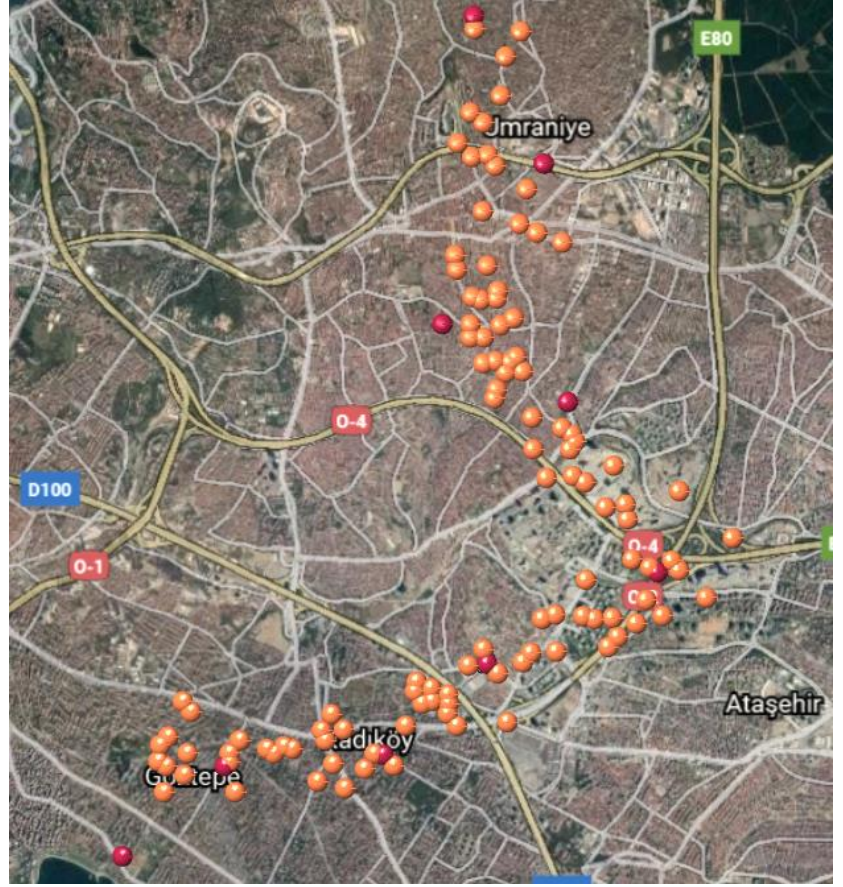
Simülasyon başladıktan sonra modelin çalışmasında bazı eksiklikler olduğu saptanmıştır. Tüm etmenler istasyonlara sırayla uğrarken aynı zamanda dağılmalar meydana gelmeye başlamıştır (Şekil 15).



Şekil 15: Etmenlerin zamana bağlı olarak ilerlemesi.

Etmenlerin zamana bağlı olarak ilerlemesinde tespit edilen dağılımları giderebilmek için öncelikle aralarında ne kadar mesafe olduğunda dağılımların yaşandığı tespit edilmiştir. Bunun için öncelikle sürüyü oluşturan etmen yoğunluğu azaltılıp Grasshopper'da bulunan operatörlerin yardımı ile gruplamalar yapılmıştır. Closest Points

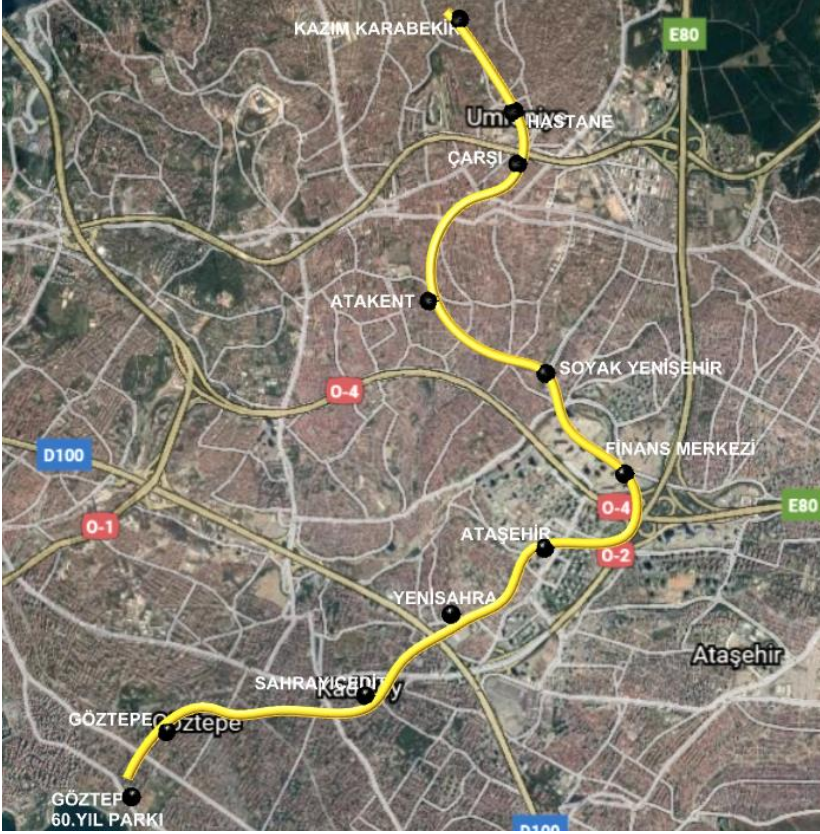
parametresi, bir nokta bulutu içerisinde belirli bir noktaya yakın olan elemanlarla grup oluşturur. Gruptaki eleman sayısı opsiyoneldir. Sürü Closest Points bileşeni ile sadeleştirilmiş ve sonrasında Smaller Than bileşeni ile aralarındaki mesafeye göre kalan etmenler tekrar gruplanmıştır. Mesafesi 4.00 birimden küçük olan etmenler simülasyonda bırakıldığında dağılımların olmadığı gözlemlenmiştir (Şekil 16).



Şekil 16: Gruplanmış etmenlerin yayılımı.

Yayılımın metro güzergahı üzerinden araştırıldığı düşünüldüğünde, anlaşılmasını kolaylaştırmak adına temsili değiştirilmiştir. Çok fazla etmen bulunması ve güzergâh incelemesi yapılabilmesi için sürü elemanları daha da sadeleştirilerek, belirlenen parametreler ile civık mantar sürüsüne tekrar güzergâh çizdirilmiştir. Smaller Than bileşeninden gelen noktalar kendi içlerinde önce gruplanmış, listesi çıkarılmış ve bu listedeki noktalardan çokgen oluşturularak, çokgenlerin merkez noktaları seçtirilmiştir. Daha sonra merkez noktaları tekrar listeye dönüştürülüp, nokta olarak çıktı vermesi sağlanmıştır.

Elde edilen daha az sayıda noktasal etmenler Interpolate Curve ile birbirine bağlanıp Pipe komutu ile kalınlık verilerek, durak noktaları arasında cıvık mantarların kendi davranış şekilleri ile oluşan metro güzergahı ortaya çıkarılmıştır (Şekil 17).



Şekil 17: Modelde üretilen metro güzergahı.

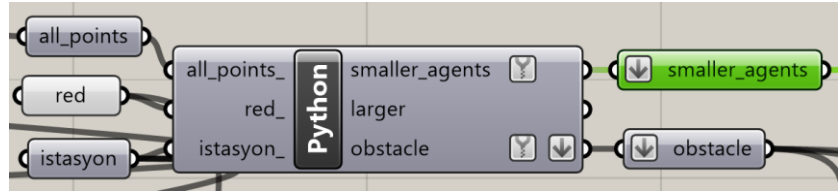
Modelde etmenlerin oluşturduğu güzergâhın geçtiği bölgelere bakıldığında, istasyon çıkışları, metro güzergahı, raylı sistem tasarım kriterleri ile beraber değerlendirildiğinde üç istasyon için modeldeki etmenlere müdahale edilmesi gerekliliği ortaya çıkmıştır:

- Göztepe durağı: Metro çıkışlarının kamusal alanlara yakınlığı söz konusu olduğu için metro güzergahının Göztepe Parkına doğru kaydırılması gerekmektedir.
- Sahrayı Cedit durağı: Entegrasyon ilkesi metro tasarımının ana ilkelerinden olduğu için Minibüs caddesindeki lastikli araç yolu ile metro güzergahının entegre olması kaçınılmaz bir durumdur.
- Yeni Sahra durağı: Metro istasyonları maliyet açısından değerlendirildiğinde, kamulaştırmanın az olduğu bölgelere yapılmaktadır. Bu nedenle Metro Marketler Zincirinin binasının

bulunduğu yerden geçiyor olması maliyeti artıracığı için güzergahın başka doğrultuya kaydırılması gerekmektedir.

Belirlenen istasyonlarda yapılacak müdahalelerin tek adımda olabilmesi için de Grasshopper içerisinde bulunan GhPython bileşeninden faydalanılmıştır. Model içerisine Python kodu eklemeye olanak sağlayan GhPython bileşeni isteğe göre artırılıp azaltılabilen iki giriş ve iki çıkış noktası bulundurmaktadır. Giriş ve çıkış noktaları yazılacak kod parçasına göre isimlendirilmektedir. Gereksiz düğüm yükü oluşturmak yerine GhPython bileşenine bağlanan girdiler ile bileşenin çıktı düğümlerinden daha kolay sonuç elde edilmiştir (**Şekil 18**).

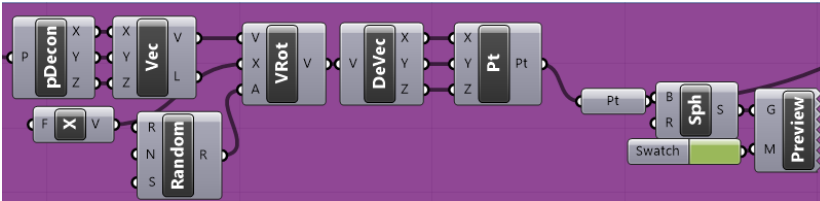
Şekil 18: Python bileşeni, girdi ve çıktıları.



Yapılan bu çalışmada mevcut güzergâh üzerinden gidildiğinde, civık mantarların laboratuvar deneylerinde alınan verimin karşılığının bilgisayar ortamında da alınabilmesi için sürü zekasının çalışma mantığının gerektirdiği kurallara ihtiyaç duyulmuştur. Etmenlerin dağılımlarını önlemek, engellerden uzaklaşmalarını sağlamak ve yönelim kuvveti uygulayabilmek için birtakım kuralların belirtilmesi gerekmiştir. Bu nedenle Python bileşeni ile belirlenen kurallar modele dahil edilmiş ve elde edilen çıktılar üzerinden model görselleştirilmiştir. Öncelikle GhPython bileşeninde yazılan kod, sürü elemanlarının birbirlerinden uzaklaşma mesafesi belirlenirken kullanılan Smaller Than bileşeni ile aynı görevi görerek; etmenlerin “noktadan listeye, listeden noktaya” dönüştürülmesi ve mesafelerin karşılaştırılmasını sağlamıştır. Bileşen içerisindeki kod ile etmenler istasyonlara uzaklıklarına göre gruplandıktan sonra, Yeni Sahra, Göztepe ve Sahrayı Cedit istasyonlarında ne gibi müdahaleler yapılacağına karar verilmiştir. Hattın Göztepe parkından ve Minibüs caddesinden geçmesinin tasarım kriterlerine daha uygun olacağı düşünüldüğü için bu bölgelere etmenlerin yönlendirilmesi gerekmektedir. Ayrıca özel mülk bölgesinden geçen güzergahın, Yeni Sahra istasyonunda o noktadan geçmemesi daha uygun olacağı için hattın oradan uzaklaştırılması gerekmektedir. Bu nedenle yönlenecek ve uzaklaşılacak bölgeler

belirlendikten sonra, bölgelere “red ve green” adlarında nokta grupları yerleştirilmiştir.

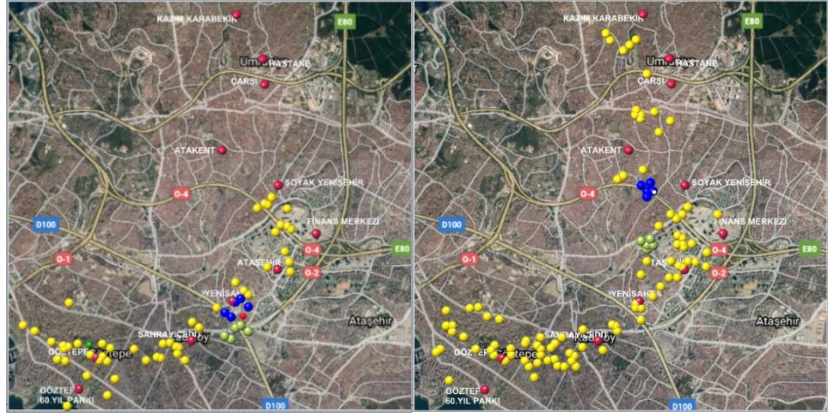
Engel noktaları için kural seti, öncelikle red noktaların yukarıda yazılan kodun düğüm noktalarından bağlanarak bileşene dahil edilmesiyle oluşturulmuştur. Bileşen içerisindeki kodun tüm etmenler ile çalıştırılarak obstacle ve smaller_agents olmak üzere iki ayrı grup şeklinde çıktı vermesi sağlanmıştır. Engel olacak red nokta bileşenin tüm etmenlere olan uzaklığı hesaplanarak obstacle çıktısı elde edilmiştir. Obstacle bileşeni içerisindeki noktalar, red noktalara olan mesafeleri 4.00 birimden az olduğu zaman rastgele olarak yönleri değiştirilerek simülasyona tekrar dahil edilmiştir. Simülasyon çalıştığı sürece kural seti tekrar tekrar devreye girerek her etmen için mesafeleri hesaplayıp yakın olanların yönünü her seferinde değiştirmektedir. Noktaların yönlerinin değiştirilebilmesi için öncelikle noktaların X, Y ve Z koordinatları çıkartılmıştır. Daha sonra koordinatlardan vektör oluşturulmuştur. Vektörler Rotate bileşenine bağlanarak; yön düğümüne Random bileşeni bağlanmıştır. Yönleri değişen vektörler tekrar koordinatlarına ayrılarak noktaya dönüştürülmüştür (**Şekil 19**). Yönleri değiştikten sonra etmenler, Physarealm kuralları ile çalışmaya devam ederek hedef istasyonlara ilerlemeye devam etmiştir. Engelden uzaklaşan etmenler dışında her seferinde doğan etmenler için de kurallar uygulanmıştır.



Şekil 19: Obstacle bileşenindeki noktaların yönlerinin değiştirilmesi.

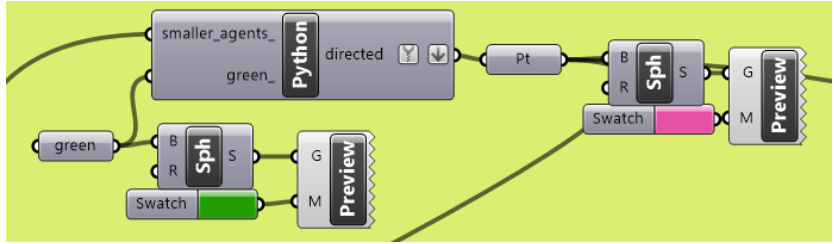
Red bileşenine bağlı engel noktalarının yeri değiştirildiğinde; her seferinde tüm etmenlerin engellere olan uzaklıkları hesaplanarak, uzaklığı 4 birimden az olan etmenlerin yönleri rastgele olarak değiştirilmiştir (**Şekil 20**). Yönü değişen etmenler yeşil renk ile temsil edilirken, değişmeden önceki etmenlerin rengi lacivert ile temsil edilmiştir. Lacivert etmenler yönlerini daha rahat tespit edilebilmesi için ayrı bir grup olarak saklanmıştır.

Şekil 20: Yönleri değişen etmenleri yayılımı.



Yönelim yapılacak noktaları içerisinde barındıran green bileşeni GhPython kodunun çıktısı olan smaller_agents' lar ile yeni bir GhPython bileşenine bağlanmıştır (Şekil 21).

Şekil 21: Yönelim için oluşturulan düğüm dizisi.



Bileşenin içerisinde bulunan kod ve işlevi aşağıdaki şekildedir:

- Çalışılacak kütüphane çağırılır.
- Düğüm noktalarından bağlanan girdiler nokta olarak tanıtlır.
- İstasyonlara olan uzaklıklarına göre etmenler gruplandıktan sonra engelle yönelinilecek noktalara olan uzaklıklar ölçülür.

“Eğer etmenlerin uzaklıkları 8.00 birimden az ise directed listesi içerisine yazdır, değil ise geri dön ve liste dışı bırak.” Uzaklıkları 8.00 den küçük olan etmenler listelenip, green noktalar ve smaller_agents noktaları arasında vektörler oluşturulup, bu vektörler vasıtası ile smaller_agents noktaları green noktalara taşınır.

Green noktalara yaklaşan her etmenin uzaklığı hesaplanıp, 8.00'den az mesafede bulunanlar her seferinde green noktalara yaklaştırılmıştır. Simülasyon çalıştığı sürece, doğan her etmen için tekrar tekrar kurallar uygulanmıştır. Directed çıktısı noktasallaştırılıp görsel olarak kürelere bağlanıp simülasyon başlatıldığında, Python kodu, smaller_agents' ların

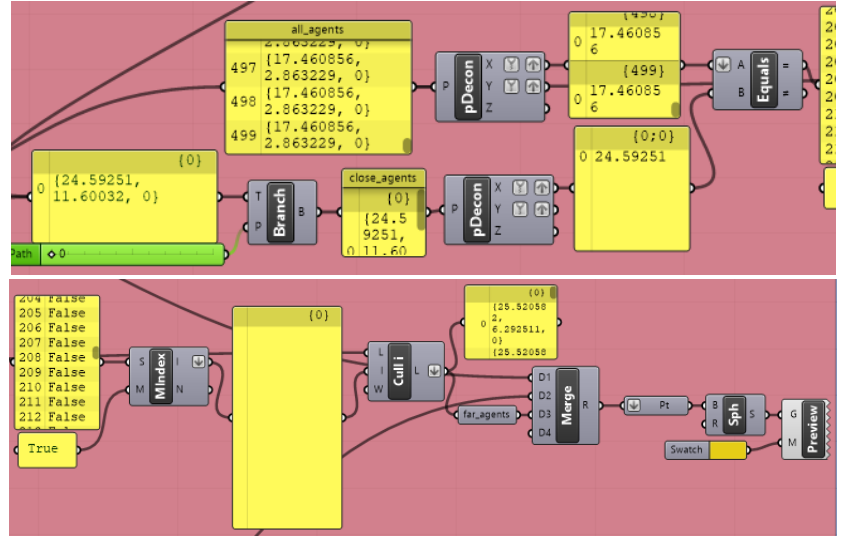
green bileşeninde tanımlı yeşil noktalara olan uzaklığı 8.00 den az olduğunda bu etmenlerin green bileşeninde tanımlı noktalara taşınmasını sağlamıştır (Şekil 22).



Şekil 22: Yönelimi yapılan etmenlerin dağılımı.

Green bileşeninde tanımlı noktalara yaklaştığında sarı renkli kürelerin rengi pembeye dönüşmektedir. Yönü değişmesi gereken etmenler, yönü değişen etmenler, yönelimi yapılacak etmenler ve yönelmiş etmenlerin tamamı son sürünün içerisinde çalıştırılmaktadır. Fakat bu son durum içerisinde, etmenler iki kere çalıştırılıp aksaklık oluşturacağı için obstacle ve directed bileşenlerinde bulunan etmenlerin, tüm etmen sürüsünden ayrıştırılması gerekmektedir. Bu nedenle tüm sürü bir "ağaç olarak düşünülüp" tekrar eden dallar bu ağaçtan çıkarılmıştır (Şekil 23).

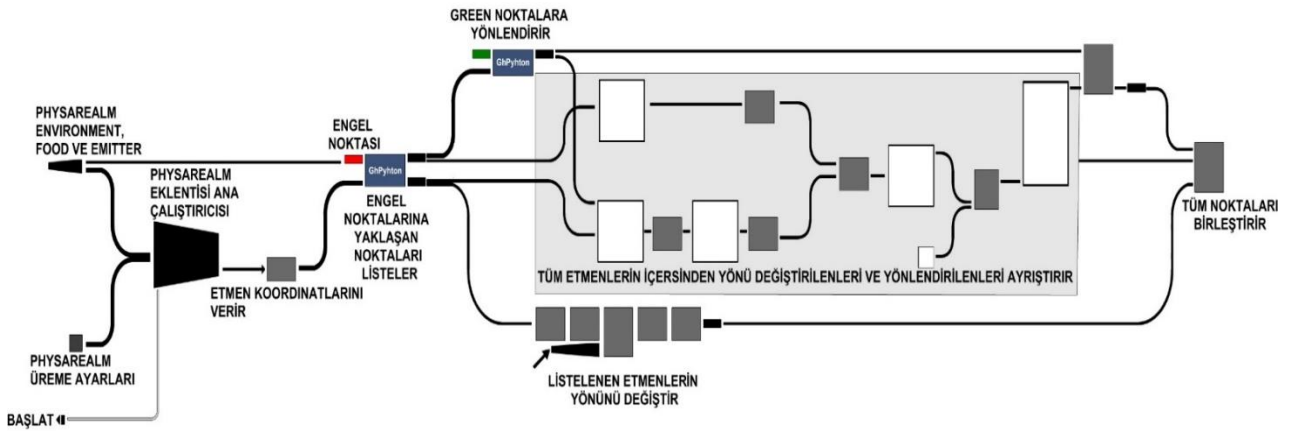
Şekil 23: Etmen Listelerinin Ayırıştırılması.



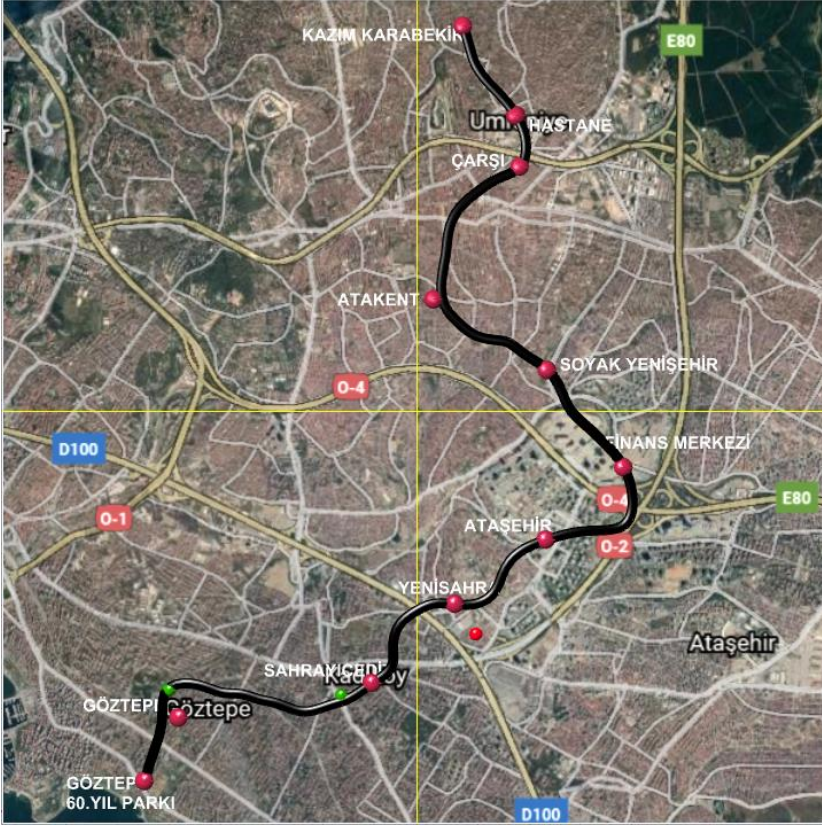
Cıvık mantar simülasyonu oluşturan etmenlerden, yönü değiştirilmiş ve belirli bir koordinata yönlendirilmiş olan etmenlerin listeleri tüm listeden çıkarıldıktan sonra, Merge komutu ile tüm etmenler tekrar tek bir listede toplanmıştır. Liste çıktısı önce noktasal olarak alınmıştır. Son durumda sürüde çok sayıda etmen bulunduğu için noktaların eğrisel olarak ifadesi Grasshopper ortamında mümkün olmadığı için etmenler, Rhinoceros ortamına yapıştırılıp, noktalar tek bir eğriye dönüştürülmüştür.

Şekil 24: Model akış diyagramı.

Etmenler ile Grasshopper içerisinde oluşturulan model sadeleştirilerek bir diyagram olarak ifade edilmiştir (Şekil 24). Çok fazla bileşen kullanılması ve bileşenlerin düğüm noktalarının fazla olmasından doğan karmaşıklığın en aza indirgenmesi için Grasshopper ortamındaki ifadelerine benzer olmasına dikkat edilmiştir.



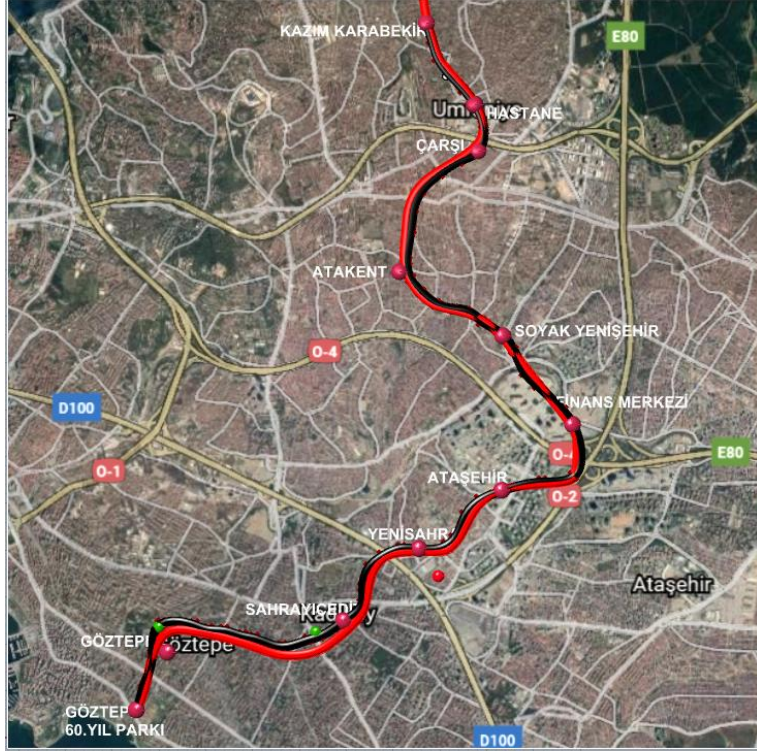
Geliştirilen model üzerinden, etmenlerin üreme parametreleri saptandıktan sonra müdahale olmaksızın metro güzergahı tasarımını gerçekleştirdikleri gözlemlenmiştir (Şekil 25).



Şekil 25: Simülasyon sonucu elde edilen metro hattı.

Canlı organizmaların belirme kavramı üzerinden şekillenen yaşam döngüleri ışığında, mevcut bir metro hattının istasyon yerlerinin bulunacağı bölgelere bağlı kalınarak metro güzergahı mevcuttan bağımsız olarak model aracılığıyla çizdirilmiştir. Belirlenen parametreler ve metro tasarım kısıtları göz önüne alınarak, ortaya çıkan güzergaha üç istasyon bölgesinde müdahale edilmesi gerektiği düşünülmüştür. Yeni Sahra istasyonundan geçtikleri noktaların uygunsuzluğundan dolayı o bölgeden uzaklaştırılırken, Sahrayı Cedit ve Göztepe istasyonlarında entegrasyon göz önüne alınarak başka bölgelere yönelimleri sağlanmıştır. Etmenlerin bu müdahaleler sonucunda oluşturdukları metro güzergahı ve mevcutta yapımı devam eden Göztepe-Ümraniye metro güzergahı arasında büyük ölçüde benzerlik olduğu gözlenmiştir (Şekil 26).

Şekil 26: Mevcut güzergâh ile model sonucu elde edilen güzergahın karşılaştırılması (Mevcut hat: kırmızı, Model hattı: siyah).



3.3 Çalışmadan Elde Edilen Çıkarımlar

Bu çalışma kapsamında yapılan literatür incelemesinde, cıvık mantarların metro hatlarının veya otoyolların doğruluklarını sınamak için kullanıldığı görülmüştür. Fakat bu deneyler genel olarak laboratuvar ortamında, agar kalıplar ve haritalar gibi fiziki materyaller ile gerçekleştirilmiştir. Physarealm eklentisi her ne kadar bu organizmaların yaşam döngüleri üzerine kurulu algoritmalar ile geliştirilmiş olsa da eklentinin bu tarz bir çalışmada kullanımına rastlanmamıştır. Bu anlamda yapılan deneylerin fiziki şartlardan bağımsız olarak geliştirilebilecek olması, hem tasarım aşamasında hem de mevcut hatların uygunluğunun sınanmasında daha özgür bir süreç sağlayacaktır.

Kentsel tasarımın önemli aktörlerinden olan ulaşım ağlarının şekillenmesi oldukça zahmetli süreçler gerektirmektedir. Çok sayıda matematik işlemi ve hesaplama yükü barındıran raylı sistem tasarımları, birçok mühendislik çalışması sonucu ortaya çıkmaktadır. Uzun hesaplama sürelerinin hem yeni bir metro hattının tasarımında hem de mevcut bir hatta ekleme veya hattan çıkarmalar yapılacağı durumlarda tasarımın hızlı olması gerekmektedir. Özellikle mevcut hatlar üzerinde

çalışıldığında gerek diğer ulaşım ağlarıyla entegre olduğu daha lokal bölgelerin verimliliği açısından, gerekse ulaşımın sürekliliğinin sağlanabilmesi için daha hızlı ama bir o kadar da güvenilir sonuçların elde edilmesi gerekir. Bu çalışmanın ışığında elde edilen veriler ile metro güzergahları tasarlanırken dijital ortamlarda yapılacak simülasyon modellerinin, zamansal ve parasal anlamda birçok kazanımlar sağlayacağı düşünülmektedir.

4. SONUÇLAR

Meta-düzey davranışların ortaya çıktığı sürü davranışı, kendini meydana getiren alt etmenlerin işbirlikçi davranışlarına dayanan etkileşimlerden oluşur. Etkileşen birimlerin çokluğu da en az etkileşimler kadar önemlidir. Karmaşık ve beliren sistemlerin temelini oluşturan bu toplu davranış biçimi, sürü zekâsı olarak adlandırılır. Bu davranış biçimi geri bildirim ve tekrar esaslıdır; birçok matematiksel modele de örnek olurken, aynı zamanda tasarım için yaratıcı olanaklar sunmaktadır. Basit yapıların daha karmaşık sistemler oluşturması, kentsel tasarım ve bu çalışmanın konusu olan metro güzergahlarının tasarımı üzerinden bakıldığında, tasarım aşamasında süreci planlarken avantaj sağlayabilmektedir. Cıvık mantarların bireysel olarak ulaşamadıkları besin kaynaklarına aralarında kemoatraktör iz ile iletişim kurarak yönelmeleri ve bu yönelimin çevresel faktörlerden oldukça etkileniyor olması tasarım sorunlarını tanımlarken de yol gösterici olabilmektedir.

Kullanıcı geribildirimlerinin ve birçok unsurun etkileşiminin önemli olduğu metro güzergahları, her ne kadar planlansa da insan unsurunu barındırdığı sürece makro düzeyde sonuçlar doğuran kentlerin önemli bileşenlerindedir. Cıvık mantarların ve kentlerin önemli beliren sistem örnekleri olması ve cıvık mantar davranışlarının metro güzergahlarının tasarımında kullanılması kentlerin planlanmasında verimli sonuçlar elde edilmesini sağlamaktadır.

Bu çalışmada, cıvık mantarların beliren davranışlarının altında yatan algoritmik temeller kurallarla tanımlanıp sürünün oluşturduğu güzergahlar incelenmiştir. Cıvık mantar etmenlerinin iletişimlerini sağlayan kemoatraktör iz miktarı bağlamında üretilen Physarealm eklentisi çalışma ortamı olarak seçilerek mantar davranışlarındaki değişimlerin gözlemlenmesini sağlamıştır. Model çıktıları, dijital ortamların, canlı organizmaların ve yaşam döngülerinin ışığında tasarım

aşamalarına yaptığı katkılar açıkça göstermektedir. Cıvık mantar sürülerinin fiziksel ihtiyaçları doğrultusunda açığa çıkan bu davranışların metro güzergahlarının tasarım mantığı ile bu denli örtüşmesi, beliren sistemlerin tasarım aşamasında birçok katkı sağlayacağını göstermektedir. Cıvık mantar davranışları referans alınarak üretilen model ile metro güzergahlarının tasarım aşamasının uzun süreler gerektiren hesaplama yükünün azaltılması sağlanabilmektedir.

Yapılan bu çalışma ile istasyon yerleri ve güzergâh uzunlukları gibi tasarım kriterleri belirlenen metro hatlarının, program üzerinde belirlenen parametreler ile hızlı bir şekilde oluşturulabileceği görülmüştür. Metro tasarımlarının daha hızlı gerçekleşmesi beliren sistemlerin önemli örneklerinden olan kentlerin oluşum hızlarına ulaşım sistemlerinin daha rahat ayak uydurmasını sağlayacaktır. Sayısal modellemenin sürekli gelişerek her ortama dahil olduğu günümüz dünyasında, biyolojik sistemlerin kendini organize eden yapıları ve bu yapıların altında yatan algoritmaların kullanımı her alanda olduğu gibi tasarımcıya üretkenlik konusunda çeşitli alternatifler sağlayacaktır.

Referanslar

Adamatzky, A., Martinez, G. J., Chapa-Vergara, S. V., Asomoza-Palacio, R., Stephens, C. R., (2010). Approximating Mexican Highways with Slime Mould. *Advances in Complex Systems* 10 (3), 1195-1214.

Adamatzky, A., Akl, S. G., (2011). Trans-canada Slimeways: Slime Mould Imitates the Canadian Transport Network. doi: 10.4018/978-1-4666-4253-9.ch017. arXiv:1105.5084.

Adamatzky, A., Yang, X., Zhao, Y., (2013). Slime Mould Imitates Transport Networks in China. *International Journal of Intelligent Computing and Cybernetics* 6 (3), 232-251.

Adamatzky, A., Jones, J., (2009). Road Planning with Slime Mould: If Physarum Built Motorways It Would Route M6/M74 through Newcastle. *International Journal of Bifurcation and Chaos* 20(10), 3065-3084. doi:10.1142/S0218127410027568.

Adamatzky, A., Martinez, G., Lees, M., Slood, P. M. A. (2012). Bio-Development of Motorway Networks in The Netherlands: A Slime Mould Approach. *Advances in Complex Systems* 16 (02-03). doi:10.1142/S0219525912500348.

Adamatzky, A., Prokopenkob, M. (2011). Slime Mould Evaluation of Australian Motorways. *International Journal of Parallel Emergent and Distributed Systems* 4, 275-295.

American Association for the Advancement of Science. (2010, January 22). Slime Design Mimics Tokyo's Rail System: Efficient Methods of a Slime Mold Could Inform Human Engineers [website]. *ScienceDaily*. Retrieved from: www.sciencedaily.com/releases/2010/01/100121141051.html

Beni, G., Wang, J. (1990). Self-Organizing Sensor Systems. *NATO ASI Series F: Computer and Systems Sciences* 58, 251-261.

Evangelidis, V., Jones, J., Dourvas, N., Tsompanas, M. A., Sirakoulis, G. Ch., Adamatzky, A. (2017). Physarum Machines Imitating a Roman Road Network: the 3D Approach. *Scientific Reports* 7(1). doi: 10.1038/s41598-017-06961-y.

Hensel, M., Menges, A., Weinstock, M. (2004). *Emergence: Morphogenetic Design Strategies*. Argentina: Wiley.

Hofstadter, D. (1979). *Godel, Escher, Bach: An Eternal Golden Braid*. New York: Basic Books.

Jones, J. (2010). Characteristics of Pattern Formation and Evolution in Approximations of Physarum Transport Networks. *Artificial Life* 16, 127-153.

Leach, N. (2009). Swarm Urbanism. *Architectural Design* 79, 56-63.

Toht, A., Nakagaki, T. (2000). Intelligence: Maze- Solving by an Amoeboid Organism. *Nature* 407, (s.470-470).

Url-1<<http://archieboytom.blogspot.com>>, erişim tarihi 15.02.2020

Url-2< <https://discovermagazine.com/>>, erişim tarihi 22.03.2020

Url-3< <https://discovermagazine.com/>>, erişim tarihi 23.03.2020

Url-4< <https://www.researchgate.net/>>, erişim tarihi 17.03.2020

Url-5< <https://www.semanticscholar.org/>>, erişim tarihi 23.03.2020

Url-6< <https://www.researchgate.net/>>, erişim tarihi 17.03.2020

Url-7< <https://www.semanticscholar.org/>>, erişim tarihi 02.03.2020

Url-8< <https://www.semanticscholar.org/>>, erişim tarihi 19.03.2020