

MATEMATİKSEL SİMÜLASYON: TEORİ, UYGULAMA VE KARAR DESTEK

$$\begin{aligned}
 & \lim_{x \rightarrow 1} \frac{ctgx-2}{2^{11}x^3} \quad \int (x \pm a)^c \quad \sum_{i=1}^n \frac{A_i - C_i}{n} \\
 & +y^2 = z \quad S_3 = \begin{bmatrix} 10 & 0 \\ 10 & 1 \\ 0 & 1 \end{bmatrix} \quad \phi = \sqrt{\frac{\sum (x-m)^2}{n-1}} \quad S = \int_2^5 f(t) dt \\
 & \pi \approx 3,1415 \quad \sin x \quad y = \frac{\Delta x}{\Delta z} \\
 & P = r^2 \pi \quad (x+y)^2 = \left(\frac{y}{2}\right)^2 \quad \frac{\Delta x}{\Delta y} = \lim_{\Delta y \rightarrow 0} \frac{\Delta x + 2}{\Delta y - 1} \\
 & \Delta t = T - \frac{3a}{x} \quad 8x = 4 - 3y^2 \quad (x+a)^2 = x^2 + 2ax + a^2 \quad f_x = \\
 & (x-y^2) \quad y = 2x^2 + 3x \quad \ln \left(\frac{a-y}{x} \right) \quad X_{1/2} = \frac{b \pm (a-c)}{\sqrt{2a}} \\
 & f = \frac{\sqrt{x+a^2}}{x} \quad e = 2,79 \quad e = \cos x + \tan y \quad \tan(2a) = \frac{2 \tan(a)}{1 - \tan^2(a)} \\
 & P = \sum_{i=0}^{\infty} X_i^e \quad y = \frac{\Delta x}{\Delta z} \quad \ln \sqrt{axb} \quad \sum_{n=0}^{\infty} \frac{x^n}{n!} \\
 & (y-1)^2 \quad \sin a = \frac{b}{c} \quad \alpha, \beta, \gamma \quad a^2 + b^2 = c^2
 \end{aligned}$$

DR. ÖĞR. ÜYESİ SAJEDEH N SIGAROODI

Matematiksel Simülasyon: Teori, Uygulama ve Karar Destek

Dr. Öğretim Üyesi Sajedah N SIGAROODI

İmtiyaz Sahibi

Platanus Publishing®

Genel Yayın Yönetmeni

Arzu Betül ÇUHACIOĞLU

Kapak & Mizanpaj & Sosyal Medya

Platanus Yayın Grubu

Birinci Basım

Nisan, 2025

Yayımcı Sertifika No

45813

Matbaa Sertifika No

47381

ISBN

978-625-7060-89-9

©copyright

Bu kitabın yayım hakkı Platanus Publishing'e aittir.

Kaynak gösterilmeden alıntı yapılamaz, izin alınmadan hiçbir yolla çoğaltılamaz.

Adres: Natoyolu Cad. Fahri Korutürk Mah. 157/B,

06480, Mamak, Ankara, Türkiye.

Telefon: +90 312 390 1 118

web: www.platanuskita.com

e-mail: platanuskita@gmail.com



PLATANUS PUBLISHING®

İçindekiler

Bölüm 1. Giriş	9
Matematiksel Modelleme ve Simülasyonun Önemi	9
1.2 Tek Disiplinli ve Çok Disiplinli Yaklaşımları	11
1.2.1 Tek Disiplinli Yaklaşım	11
1.2.2 Çok Disiplinli Yaklaşım.....	11
1.2.3 Çok Disiplinli Yaklaşımın Avantajları	12
Bölüm 2: Temel Kavramlar	12
2.1 Modelleme Türleri	13
2.1.1 Deterministik Modeller.....	13
2.1.2 Stokastik Modeller.....	14
2.1.3 Deterministik ve Stokastik Modellerin Karşılaştırılması.....	14
2.2 Simülasyon Türleri	15
2.2.1 Ayrık Simülasyon (Discrete Simulation)	15
2.2.2 Sürekli Simülasyon (Continuous Simulation)	16
2.2.3 Hibrit Simülasyon (Hybrid Simulation)	16
2.3 Parametrelerin Belirlenmesi ve Doğrulama Sureci.....	17
2.3.1 Parametrelerin Tanımlanması ve Seçimi.....	17
2.3.2 Model Doğrulama (Verification)	18
2.3.3 Model Geçerliliği (Validation)	18
2.4 Hassasiyet Analizi (Sensitivity Analysis).....	20
2.4.1 Hassasiyet Analizinin Tanımı ve Önemi.....	20
2.4.2 Hassasiyet Analizinde Kullanılan Yöntemler	20
2.4.3 Hassasiyet Analizinin Uygulaması	21
2.4.4 Multidisipliner Uygulama Alanları	21
2.5 Karmaşık Sistemler ve Sistem Teorisi.....	22
2.5.1 Karmaşık Sistemlerin Modellenmesi ve Örnekler.....	23
2.5.2 Karmaşık Sistemlerin Önemi ve Uygulamaları.....	24

Bölüm 3. Simülasyon Teknikleri	25
3.1 Ayrık Olay Simülasyonu (DES)	25
3.2 Süreç Tabanlı Simülasyon (Process-Based Simulation)	28
3.3 Sistem Dinamiği	32
3.3.1 Temel Kavramlar ve Modelleme Süreçleri	32
3.4 Monte Carlo Simülasyon (MCS)	37
3.5 Ajan-Tabanlı Simülasyon (ABM: Agent Based Modeling)	39
3.5.1 Ajan Tabanlı Simülasyonun Temel Unsurları	39
3.5.2 Ajan Tabanlı Simülasyonun Avantajları	40
3.6 Kural tabanlı simülasyon	45
3.6.2 Kural Tabanlı Simülasyon: Trafik Yönetimi Üzerine Bir Uygulama	46
3.7 Hibrit Simülasyon Yaklaşımları	48
3.7.1 Hibrit Simülasyonun Temelleri	48
Bölüm 4. Simülasyon Modeli Seçim Rehberi	54
4.1 Problemi Anlamak ve Kapsam Belirlemek	54
4.2 Veri Yapısı ve Modelleme Gereksinimleri	57
4.2.1 Veri ve Modelleme İlişkisi	58
4.3 Simülasyon Modeli Seçiminde Karşılaştırma Kriterleri	59
4.3.1 Sistem Yapısı ve Model Uygunluğu	59
4.3.2 Modelin Matematiksel Karmaşıklığı ve Hesaplama Gücü Gereksinimi.	60
4.3.3 Modelin Veri Uygunluğu ve Geçerliliği	61
4.4 Karar Verme Sürecinde Kullanılan Yöntemler (AHP, DEMATEL vb.)	61
4.5 Seçim Sürecinde Yaygın Hatalar ve Çözümleri	75
Bölüm 5. Simülasyon Yazılımları ve Araçları	79
5.1 Simülasyon Yazılımları	79
5.2 Simülasyon Yazılımlarının Karşılaştırılması	82
5.3 Simülasyon Yazılımları ile Uygulama Örnekleri	84
5.4 Yazılım Seçiminde Kriterler	94

Bölüm 6. Gerçek Dünya Projeleri ve Simülasyon Uygulamaları	96
6.1 örnek 1: Türkiye’de Sel Tahminleri için Simülasyon Kullanımı	97
6.2 örnek 2: Gazete Satıcısı Probleminin Simülasyonu	101
6.3 Örnek 3: Chicago O'Hare Bagaj Taşıma Sistemi	108
6.4 Örnek 4: Hastane Simülasyonu	110
Bölüm 7. Simülasyon Projelerinde Başarı İçin Kılavuz	122
7.1 Simülasyon Proje Adımları	122
7.2 Model Kavramsallaştırma	125
7.3 Veri Toplama	127
7.5 Doğrulama ve Geçerleme	129
7.6 Analiz	130
7.7 Dokümantasyon	131
7.8 Uygulama	133
Bölüm 8. Gelecek Perspektifleri	135
8.1 Simülasyonda Yapay Zekâ ve Makine Öğrenmesi	135
8.1.1 Yapay Zekâ ve Makine Öğrenmesinin Simülasyona Katkıları	135
8.1.2 Simülasyonda Kullanılan Yapay Zekâ ve Makine Öğrenmesi Teknikleri	136
8.1.3 Yapay Zekâ Destekli Simülasyonun Uygulama Alanları	137
8.1.4 Gelecek Perspektifi ve Karşılaşılan Zorluklar	137
8.2 Dijital İkizler ve Endüstri 4.0	138
8.2.1 Dijital İkiz Kavramı ve Temel Prensipleri	138
8.2.2 Endüstri 4.0 ile Dijital İkizlerin Entegrasyonu	139
8.2.3 Dijital İkizlerin Uygulama Alanları	139
8.2.4 Dijital İkizlerin Karşılaştığı Zorluklar ve Gelecek Perspektifi	140
8.3 Multidisipliner Simülasyonun Geleceği	141
8.3.1 Multidisipliner Simülasyonun Tanımı ve Temel İlkeleri	141
8.3.3 Multidisipliner Simülasyonun Uygulama Alanları	143
8.3.4 Multidisipliner Simülasyonun Geleceği ve Karşılaşılan Zorluklar	143

8.4 Kuantum Bilişim ve Simülasyon	144
8.4.1 Kuantum Bilişimin Temel İlkeleri ve Simülasyon ile İlişkisi.....	145
8.4.2 Kuantum Simülasyonun Avantajları	145
8.4.3 Kuantum Simülasyonun Uygulama Alanları.....	146
8.4.4 Kuantum Simülasyonun Geleceği ve Karşılaşılan Zorluklar.....	147
8.5 Etik ve Sürdürülebilirlik	147
8.5.1 Simülasyonda Etik İlkeler	148
8.5.2 Sürdürülebilir Simülasyon ve Çevresel Etkiler.....	149
8.5.3 Sosyal Sürdürülebilirlik ve Simülasyon.....	149
8.5.4 Etik ve Sürdürülebilir Simülasyon İçin Gelecek Perspektifleri.....	150
Son Söz	153
Kaynak.....	154

Bilimsel arařtırmaların ve teknolojik ilerlemelerin hızla artması, giderek daha karmařık sistemlerin anlařılmasını ve yönetilmesini zorunlu kılmaktadır. Bu tür karmařık sistemler, farklı disiplinlerin bilgi birikimiyle modellenip analiz edilebilir. Ayrık olay simülasyonu, ajan tabanlı modelleme ve sistem dinamięi gibi yöntemler, bu sistemlerin davranıřlarını anlamak ve tahmin etmek için güçlü araçlar sunmaktadır.

Bu kitap, çok disiplinli simülasyon ve modelleme yaklařımlarını hem teorik hem de uygulamalı bir perspektifle ele almayı hedeflemektedir. Matematik, mühendislik, biyoloji, ekonomi ve sosyal bilimler gibi çeřitli alanlardan örneklerle, modellemenin çok yönlülüęünü ve gücünü vurgulamaktadır.

Bu kitabın amacı, okuyuculara matematiksel modelleme ve simülasyonun temel kavramlarını, çok disiplinli sistemlerin modellenmesi için kullanılan yöntemleri ve gerçık dünyadaki karmařık problemlerin çözümüne yönelik uygulamaları tanıtmaktır.

Bu kitap, ařaęıdaki gruplar için hazırlanmıřtır:

Üniversite düzeyinde modelleme ve simülasyon öęrenmek isteyen öęrenciler,

Arařtırmalarında disiplinler arası yaklařım geliřtirmek isteyen akademisyenler,

Karmařık sistemleri anlamak ve yönetmek isteyen profesyoneller.

Matematiksel modelleme, yalnızca bir teori deęil, aynı zamanda dünyayı anlamanın ve geleceęi şekillendirmenin de bir yoludur. Bu kitap, modellemenin gücünü keřfetmek ve yeni ufuklara yelken açmak isteyen herkes için bir rehber nitelięi taşımaktadır.

Bölüm 1. Giriş

Matematiksel Modelleme ve Simülasyonun Önemi

Matematiksel modelleme, gerçek dünyadaki sistemleri ve süreçleri anlamak için soyutlama yapmayı ve bu soyutlamaları matematiksel ifadelerle temsil etmeyi amaçlar. Modelleme, fiziksel, biyolojik, ekonomik ve sosyal süreçler gibi karmaşık sistemlerin analiz edilmesine olanak tanır. Örneğin, bir nehrin akış rejimini analiz etmek için kullanılan matematiksel modeller, yalnızca suyun hareketini değil, aynı zamanda akışın çevreye olan etkilerini ve bu etkilerin gelecekteki senaryolarını da tahmin edebilir. Bunun gibi uygulamalar, matematiksel modelleme yöntemlerinin, yalnızca mevcut durumları anlamakla kalmayıp, gelecekte karşılaşılabilecek sorunlara hazırlıklı olmayı da sağladığını göstermektedir.

Modelleme, sistemlerin temel dinamiklerini anlamamıza olanak tanırken, simülasyon bu dinamiklerin nasıl çalıştığını dijital bir ortamda test etme imkânı sunar. Simülasyon, teorik modellerin uygulamada nasıl sonuçlar doğuracağını incelemek için kullanılan etkili bir yöntemdir. Özellikle mühendislikte, karmaşık makinelerin veya yapıların dayanıklılığını ölçmek amacıyla yapılan simülasyonlar, prototip üretim maliyetlerini ve riskleri önemli ölçüde azaltır. Örneğin, uçak tasarımında kullanılan simülasyonlar, rüzgâr tüneli testlerini dijital ortama taşıyarak daha hızlı ve ucuz sonuçlar elde edilmesine olanak sağlar. Benzer şekilde, otomobil sektöründe çarpışma testleri artık çoğunlukla simülasyonlarla gerçekleştirilmektedir ve bu durum hem maliyetleri düşürmekte hem de araç güvenliğini artırmaktadır.

Matematiksel modelleme ve simülasyon, sadece mühendislik veya fizik gibi disiplinlerde değil, biyoloji ve sosyal bilimlerde de önemli bir yere sahiptir. Biyolojide kullanılan modeller, bir organizmanın içindeki hücresel süreçleri anlamamızı sağlar. Örneğin, kanser hücrelerinin yayılma dinamiklerini modelleyen bir matematiksel ifade, ilaç geliştirme süreçlerini hızlandırabilir. Benzer şekilde, bir toplumda salgın hastalıkların yayılma hızını tahmin eden epidemiyolojik modeller, sağlık politikalarının belirlenmesinde hayati bir rol oynar. Bu modeller, simülasyonlarla desteklenerek bir hastalığın yayılma hızına göre karantina

önlemlerinin ne kadar etkili olacağına dair bilimsel öngörülerde bulunabilir.

Sosyal bilimlerde ise matematiksel modelleme ve simülasyon, toplumsal davranışların ve hareketlerin anlaşılmasında kullanılır. Örneğin, şehir planlamasında kullanılan trafik akış modelleri, bir bölgedeki araç yoğunluğunu azaltmaya yönelik stratejilerin belirlenmesine yardımcı olur. Ajan-tabanlı modelleme yöntemleriyle bireylerin davranışları simüle edilerek, bu davranışların toplumsal düzeyde nasıl sonuçlara yol açabileceği öngörülebilir. Bu, sadece şehirlerin daha verimli bir şekilde tasarlanmasına değil, aynı zamanda sosyal adaletin sağlanmasına yönelik politikaların geliştirilmesine de katkıda bulunur.

Matematiksel modelleme ve simülasyon, özellikle çok disiplinli yaklaşımlar gerektiren durumlarda benzersiz bir araçtır. Bir problem tek bir bilim dalının sınırları içinde çözülemeyecek kadar karmaşık olduğunda, farklı alanlardan gelen bilgiler bir araya getirilir. Örneğin, bir iklim değişikliği modelini düşünelim. Bu tür bir model, atmosfer bilimi, okyanus bilimi, biyoloji ve ekonomi gibi farklı disiplinlerden gelen verilerin bir arada kullanılmasıyla oluşturulur. Böylece, bu tür modeller, iklim değişikliğinin çevresel ve ekonomik sonuçlarını anlamamıza ve gerekli önlemleri belirlememize yardımcı olur.

Simülasyonların sağladığı bir diğer avantaj da sistemlerin gelecekteki davranışlarını öngörebilme yeteneğidir. Örneğin, enerji şebekelerinin davranışını modellemek, gelecekteki enerji taleplerine göre şebekeleri optimize etmeye olanak tanır. Bu tür simülasyonlar, yenilenebilir enerji kaynaklarının entegrasyonu gibi karmaşık sorunların üstesinden gelmek için de kullanılmaktadır. Matematiksel modelleme ve simülasyon, karmaşıklığı yönetilebilir bir seviyeye indirerek hem bilimsel hem de pratik anlamda büyük bir değer sunar.

1.2 Tek Disiplinli ve Çok Disiplinli Yaklaşımları

1.2.1 Tek Disiplinli Yaklaşım

Tek disiplinli yaklaşım, bir problemi çözmek için yalnızca bir akademik disiplinin bilgi, yöntem ve bakış açılarını kullanmayı ifade eder. Bu yaklaşımda, çözülmesi gereken problem bir disiplinin uzmanlık alanına ait olan teoriler ve araçlar doğrultusunda ele alınır. Örneğin, bir fiziksel sistemi modellemek amacıyla sadece fiziksel yasalar kullanılır ve çözüm süreci bu alanın kuralları çerçevesinde gerçekleştirilir. Tek disiplinli yaklaşımlar genellikle belirli bir konuya derinlemesine odaklanmayı mümkün kılar, fakat çözümün diğer etmenlerle olan ilişkileri genellikle göz ardı edilir.

Tek disiplinli yaklaşımlar, belirli bir disiplinde uzmanlaşmak ve o alandaki derin bilgiyi uygulamak için güçlüdür. Ancak bu tür bir yaklaşım, özellikle modern dünyada, karmaşık ve birbirine bağlı sorunları ele alırken eksikliklere yol açabilir çünkü yalnızca bir perspektiften bakar.

1.2.2 Çok Disiplinli Yaklaşım

Çok disiplinli yaklaşım, bir problemi çözmek için birden fazla disiplinin bilgi ve yöntemlerini bir araya getiren bir yaklaşımdır. Bu yöntemde, her bir disiplin kendi uzmanlık alanını koruyarak, ortak bir problemin çözümüne katkıda bulunur. Çok disiplinli yaklaşımlar, farklı disiplinlerin birbirinden bağımsız olarak katkı sağladığı ve disiplinler arasındaki etkileşimin minimal olduğu bir iş birliği biçimini ifade eder. Örneğin, şehir planlamasında, mühendislik, sosyoloji ve ekonomi gibi farklı disiplinlerin bilgi ve yöntemleri, ortak bir hedefe yönlendirilir. Bu yaklaşımla, her bir disiplin kendi bakış açısını problem üzerinde uygular, fakat sorun tüm disiplinlerin katılımıyla çözülür.

Çok disiplinli çalışmalar, daha geniş bir perspektif sunarak problemlerin farklı boyutlarını ele almamıza yardımcı olur. Bununla birlikte, disiplinler arasındaki etkileşim sınırlı olduğundan, bazen çözümde kullanılan bilgi ve yöntemler arasında tam bir entegrasyon sağlanamayabilir.

1.2.3 Çok Disiplinli Yaklaşımın Avantajları

Çok disiplinli yaklaşım, özellikle karmaşık ve çok yönlü problemleri ele alırken önemli avantajlar sağlar. Öncelikle, farklı disiplinlerin perspektiflerini birleştirerek, daha kapsamlı ve entegre çözümler üretir. Bir problemi birden fazla açıdan ele almak hem derinlemesine hem de geniş çapta bilgi edinmeyi mümkün kılar. Örneğin, iklim değişikliği gibi küresel bir sorun, yalnızca çevre bilimiyle değil, aynı zamanda ekonomi, politika, sosyoloji gibi farklı disiplinlerle ele alınarak daha etkili çözümler üretilebilir.

Çok disiplinli yaklaşımın bir diğer avantajı ise yenilikçi çözümler üretilmesine olanak tanımasıdır. Farklı disiplinlerden gelen bilgi ve yöntemler, daha önce gözlemlenmemiş ilişkiler ve çözüm yolları keşfetmeyi sağlar. Bu, özellikle teknolojinin ve bilimsel düşüncenin hızla geliştiği alanlarda büyük bir avantajdır.

Ayrıca, çok disiplinli çalışmalar, karmaşık sistemleri daha iyi anlamamıza yardımcı olur. Tek bir disiplinin sınırları içinde ele alınamayacak kadar karmaşık olan problemler, çok disiplinli yaklaşımlar sayesinde daha net bir şekilde analiz edilebilir ve yönetilebilir. Bu yaklaşım, çeşitli alanlardan gelen uzmanların birlikte çalışmasıyla riskleri azaltabilir ve daha sürdürülebilir çözümler geliştirilmesini sağlar.

Çok disiplinli yaklaşımın bir diğer avantajı ise toplumsal ve çevresel etkilerin dikkate alınarak daha sürdürülebilir çözümler üretilmesidir. Örneğin, bir enerji projesi planlanırken sadece mühendislik açıları değil, aynı zamanda çevresel, ekonomik ve toplumsal etkiler de göz önünde bulundurulur. Bu şekilde, projelerin gelecekteki etkileri daha iyi tahmin edilebilir ve toplumun geniş kesimlerini etkileyen kararlar daha etkili bir şekilde alınabilir.

Sonuç olarak, çok disiplinli yaklaşım, tek bir disiplinin sınırlarını aşarak daha derinlemesine, yenilikçi ve entegre çözümler üretmek için güçlü bir araçtır. Bu yaklaşım, özellikle günümüzün küresel ve karmaşık sorunlarını çözme konusunda vazgeçilmez bir yöntem haline gelmiştir.

Bölüm 2: Temel Kavramlar

2.1 Modelleme Türleri

Matematiksel modelleme, karmaşık sistemleri anlamak ve tahmin etmek için temel araçlardan biridir. Modelleme sürecinde kullanılan yöntemler, modellenen sistemin doğasına bağlı olarak değişir. Bu bağlamda, modeller genellikle deterministik ve stokastik olmak üzere iki ana türe ayrılır. Her bir modelleme yöntemi, farklı sistemlere uygun avantajlar sunar ve belirli varsayımları içerir.

2.1.1 Deterministik Modeller

Deterministik modellerde, bir sistemin belirli başlangıç koşulları altında gelecekteki durumu kesin olarak öngörülebilir. Bu modeller, genellikle diferansiyel denklemler gibi matematiksel araçlarla ifade edilir. Sistemin giriş değişkenleri bilindiğinde, sonuçlar sabit ve tek anlamlıdır. Deterministik modeller, fiziksel süreçlerin simülasyonu gibi öngörülebilirlik gerektiren alanlarda sıklıkla tercih edilir.

Örnek:

Bir hareket problemi düşünelim. Bir nesnenin ivmesinin sabit olduğunu varsayalım ve bu nesnenin t anındaki konumunu hesaplamak isteyelim. Newton'un hareket denklemi şu şekilde yazılabilir;

$$x(t) = x_0 + v_0 t + \frac{1}{2} a t^2$$

Burada x_0 başlangıç konumu, v_0 başlangıç hızı, a ivme ve t zamandır.

Eğer başlangıç verileri $x_0 = 0$, $v_0 = 5 \text{ m/s}$ ve $a = 2 \text{ m/s}^2$ olarak verilirse, bu denklemi kullanarak herhangi bir t anında nesnenin konumunu kesin olarak hesaplayabiliriz.

Deterministik modellerin avantajı, sistemin tamamen öngörülebilir olmasıdır. Ancak, gerçek dünya uygulamalarında her zaman rastgelelikten ve belirsizliklerden arınmış sistemlerle çalışmak mümkün değildir. Bu nedenle, deterministik modeller bazen eksik ya da yetersiz kalabilir.

2.1.2 Stokastik Modeller

Stokastik modeller, sistemdeki belirsizlikleri ve rastgele süreçleri dikkate alan matematiksel modellerdir. Bu modeller, olasılık teorisi ve istatistiksel yöntemlere dayanır. Stokastik bir modelde, aynı başlangıç koşulları altında bile farklı sonuçlar elde edilebilir. Bu tür modeller, özellikle biyoloji, finans ve epidemiyoloji gibi rastgele değişkenlerin önemli olduğu alanlarda kullanılır.

Örnek:

Bir stokastik süreç olarak radyoaktif bir maddenin bozulmasını düşünelim. Radyoaktif atomların birim zaman diliminde bozulma olasılığı sabit bir oranla ifade edilir ve bu durum Poisson dağılımı ile modellenebilir. Belirli bir zaman diliminde K bozulmanın gerçekleşme olasılığı şu şekilde hesaplanır.

$$P(X = K) = \frac{(\lambda t)^K e^{-\lambda t}}{K!}$$

Burada λ bozulma oranı (birim zamanda beklenen bozulma sayısı), t zaman ve K bozunan atom sayısıdır.

Eğer $\lambda = 5 \text{ s}^{-1}$ ve $t = 2 \text{ s}$ verilirse, iki saniyede tam 8 atomun bozulma olasılığı şu şekilde hesaplanabilir.

$$P(X = 8) = \frac{(10)^8 e^{-10}}{8!}$$

Stokastik modeller, bu tür belirsizlikleri gerçekçi bir şekilde temsil eder ve rastgele süreçlerin etkilerini anlamamızı sağlar. Ancak, bu modellerin çözümü daha karmaşık hesaplamalar ve daha fazla veri gerektirir.

2.1.3 Deterministik ve Stokastik Modellerin Karşılaştırılması

Deterministik ve stokastik modellerin kullanım alanları, modellenen sistemin doğasına bağlı olarak değişiklik gösterir. Deterministik modeller, basit ve iyi tanımlanmış sistemler için etkili bir araçtır. Ancak, stokastik modeller belirsizlik ve rastgelelik içeren sistemleri ele almak için daha uygundur.

Bir su rezervuarının doluluk seviyesini tahmin etmeyi düşünelim. Eğer suya giriş ve çıkış miktarları tamamen biliniyorsa ve değişkenlik yoksa deterministik bir model kullanmak yeterli olabilir. Ancak, giriş miktarının hava durumuna bağlı olarak değiştiği ve rastgele dalgalanmalar olduğu bir senaryoda stokastik model tercih edilir.

Stokastik modellerin bir diğer avantajı, sistemin rastgele doğasını anlamamıza yardımcı olmasıdır. Ancak deterministik modellerin sadeliği ve hesaplama kolaylığı, bu tür bir karmaşıklığa ihtiyaç duyulmayan durumlarda önemli bir avantaj sağlar. Bu iki modelleme türü, çoğu zaman birbirini tamamlayıcı nitelikte kullanılır.

2.2 Simülasyon Türleri

Simülasyon, sistemlerin davranışlarını anlamak, analiz etmek ve tahmin etmek için kullanılan bir yöntemdir. Simülasyon türleri, modellenen süreçlerin doğasına ve zamanın ele alınış biçimine göre farklı kategorilere ayrılır. Genel olarak, ayrık (discrete), sürekli (continuous) ve hibrit (hybrid) simülasyonlar olarak üç ana başlık altında ele alınır. Bu türlerin her biri, modellemenin ve simülasyonun nasıl bir sistem üzerinde çalıştığını anlamamıza yardımcı olur. Aşağıda, her türün tanımları ve bir örnekte modelleme ile simülasyonun nasıl ayırt edilebileceği açıklanmıştır.

2.2.1 Ayrık Simülasyon (Discrete Simulation)

Ayrık simülasyon, sistemin belirli zaman noktalarında gerçekleşen olaylara odaklanır. Sistem, olaylar arasında değişmeden kalır ve yalnızca bir olay meydana geldiğinde durum değişikliği olur. Bu tür simülasyonlar, genellikle sıra, stok yönetimi veya üretim hatları gibi sistemlerde kullanılır. Buna göre, modelleme sistemin olaylara göre durum değişikliklerini tanımlayan matematiksel veya kavramsal bir model oluşturken, simülasyon model üzerinde belirli senaryolar çalıştırılarak sistemin zaman içindeki dinamikleri inceler.

Örnek:

Bir banka şubesindeki müşteri kuyruk yönetimi aşağıdaki gibidir.

Bir bankada, müşterilerin geliş zamanları bir Poisson dağılımı ile modellenir. Her müşterinin işlem süresi ise bir üstel dağılımla ifade edilir.

Sistemdeki olası durumlar (boş kasa, işlem yapılan kasa, kuyruk) matematiksel olarak tanımlanır ve simülasyon 100 müşterinin bir saat içinde bankaya gelmesi durumunda, kasaların ne kadar süreyle dolu olacağını, kuyrukta bekleyen müşterilerin ortalama bekleme süresini ve kasiyerlerin verimliliğini hesaplayan bir işlemdir.

2.2.2 Sürekli Simülasyon (Continuous Simulation)

Sürekli simülasyon, sistemin zamana bağlı olarak sürekli değiştiği durumlarda kullanılır. Özellikle fiziksel ve kimyasal süreçlerin modellenmesinde tercih edilir. Sistem, diferansiyel denklemlerle ifade edilir ve çözüm genellikle zamanın küçük aralıklarına bölünerek hesaplanır. Buna göre, modelleme Sistemin sürekli değişimini ifade eden matematiksel denklemler oluşurken simülasyon bu denklemleri zaman içinde çözülerek sistemin davranışı inceler.

Örnek:

Bir kimyasal reaksiyondaki sıcaklık değişimini incelemek için ilk olarak matematiksel modelleme ile reaksiyon sırasında sıcaklık değişimini ifade eden bir enerji dengesi denklemini oluşturuyoruz

$$\frac{dT}{dt} = \frac{-\Delta H \cdot R}{\rho C_p}$$

Burada T sıcaklık, $-\Delta H$ reaksiyon entalpisi, R reaksiyon hızı, ρ yoğunluk ve C_p ısıl kapasiteyi temsil eder.

Sonra, simülasyon olarak ise, reaksiyon başladığında, bir saatte sıcaklığın nasıl değiştiği Euler metodu gibi sayısal bir yöntemle çözülerek analiz ederiz.

2.2.3 Hibrit Simülasyon (Hybrid Simulation)

Hibrit simülasyon hem ayırık hem de sürekli simülasyonların özelliklerini birleştirir. Sistem hem sürekli değişimlere hem de belirli zaman noktalarında meydana gelen olaylara sahipse hibrit bir yaklaşım gerekir. Bu tür simülasyonlar, karmaşık sistemlerin modellenmesinde kullanılır. Buna göre, modelleme ayırık ve sürekli süreçlerin birbirine nasıl bağlandığını tanımlayan karma bir model oluşurken simülasyon ayırık ve sürekli bileşenlerin dinamik etkileşimlerini analiz eder.

Örnek:

Bir hidroelektrik santralin enerji üretimini ele alalım. Matematik modelleme ile su seviyesinin sürekli değişimi Bernoulli denklemiyle aşağıdaki gibi gösterelim.

$$h = \frac{P}{\rho \cdot g}$$

Burada h su seviyesinin yüksekliği(m), P suyun basıncı (pa), ρ su yoğunluğu ve g yerçekimi ivmesidir.

Aynı zamanda türbinin çalışıp durma anları (örneğin bakım gereksinimleri) ayrık olaylar olarak tanımlanır. Bu bilgilere göre simülasyon kullanarak bir yıl boyunca yağış miktarına bağlı olarak su seviyesinin sürekli değişimi ve bu değişim sırasında türbinin çalışmadığı durumların sisteme etkisini analiz edebiliriz.

Bu simülasyon türlerinin anlaşılması, farklı disiplinlerdeki problemlerin çözümünde hangi yöntemin uygun olacağını belirlemek açısından kritik öneme sahiptir. Modelleme, sistemin yapısının soyut bir temsilini sunarken, simülasyon, bu yapıyı test etme ve analiz etme aracıdır.

2.3 Parametrelerin Belirlenmesi ve Doğrulama Süreci

Modelleme ve simülasyon süreçlerinde parametrelerin doğru belirlenmesi ve modellerin hem doğruluk hem de geçerlilik açısından kontrol edilmesi, elde edilen sonuçların güvenilirliği için kritik öneme sahiptir. Bu bölümde, parametre seçimi, model doğrulama (verification) ve model geçerliliği (validation) süreçlerini detaylı bir şekilde ele alacak, gerçek hayattan örneklerle bu süreçleri derinlemesine inceleyeceğiz.

2.3.1 Parametrelerin Tanımlanması ve Seçimi

Bir model veya simülasyonda kullanılan parametreler, modelin tanımladığı sistemin dinamiklerini, sınır koşullarını ve girdilerini temsil eder. Bu parametrelerin yanlış seçilmesi, modelin gerçek sistemi yansıtamamasına neden olabilir. Parametrelerin belirlenmesi için genelde üç farklı yöntem kullanılmaktadır, deneysel veriler, uzman görüşleri ve teorik modeller. Deneysel veriler yönteminde, sistemden geçmişte toplanmış veriler kullanılarak parametreler belirlenir. Bu yöntem, özellikle

veri açısından zengin sistemlerde oldukça etkili bir yoldur. Ancak, bazı durumlarda veri bulunamayabilir veya sistem yeni tasarlanıyor olabilir. Bu durumda uzmanların tecrübelerine başvurulur (uzman görüşleri). Teorik modellerde ise, fiziksel veya biyolojik süreçlerin teorik olarak tanımlanır ve matematiksel modeller doğrudan parametreleri belirleyebilir. Aşağıda her biri için bir örnek çözüm verilmektedir.

Örnek: Bir fabrikadaki makine arızalarının sıklığını modellemek istiyoruz. Geçmiş bakım kayıtlarına göre makinenin her 200 saatte bir arıza yaptığı bilgisine ulaşırız. Bu durumda, arıza sıklığı parametresi olarak $\lambda = 200\text{saat}$ belirlenir.

Örnek: Bir hastanede hasta bekleme sürelerini modelleyeceğimizi varsayalım. Ancak geçmiş verilere erişimimiz yok. Acil servis uzmanları, her 30 dakikada bir 5 hasta geldiğini belirtiyor. Bu bilgiyle hasta geliş hızı $\lambda = \frac{5}{30} \text{ kii/dakika}$ olarak tanımlanır.

Örnek: Bir su deposundaki sıvı seviyesinin değişimini modellemek için Bernoulli denklemi kullanabiliriz. Denklemin parametreleri, suyun yoğunluğu ρ yerçekimi ivmesi g ve suyun basıncı P ile belirlenir.

2.3.2 Model Doğrulama (Verification)

Model doğrulama, bir modelin veya simülasyonun tasarlandığı gibi çalışıp çalışmadığını test etme sürecidir. Amaç, modelin teknik hatalarını belirlemek ve düzeltmektir. Doğrulama süreci iki adımla yapılır, kod hatalarının giderilmesi ve matematiksel doğruluk testleri. Bu süreçte, modelleme sırasında yapılan mantık veya kodlama hatalarının giderilmesi gerekir ve modelin sonuçları, teorik olarak beklenen sonuçlarla kıyaslanır. Örneğin, bir trafik simülasyonunda araç hızlarının negatif değerlere sahip olmaması gerekir.

2.3.3 Model Geçerliliği (Validation)

Model geçerliliği, modelin gerçek dünya sistemini ne kadar iyi temsil ettiğini değerlendirir. Bu süreçte model çıktıları, sistemin gerçek davranışları ile karşılaştırılır. Bu süreçte model sonuçları geçmiş veriyle karşılaştırılarak geçerlilik analiz edilir, model sonuçlarının mantıklı olup olmadığı uzmanlar tarafından incelenir ve model parametrelerinde yapılan küçük değişimlerin sonuçlara etkisi analiz edilir.

Örnek: Bir şehirdeki günlük elektrik tüketimini tahmin etmeye yönelik modelleme ve simülasyon tasarlıyoruz. Burada kullanılan tablo, simülasyon sonuçları ile gerçek gözlemlenen verileri karşılaştırarak modelin doğruluğunu değerlendiriyoruz. (Bu örnek, model geçerliliği ve doğrulama sürecini göstermek için verilmiştir)

Senaryo: Bir şehirde, belirli bir hafta için elektrik tüketimini tahmin etmek istiyoruz. Model, şehirdeki enerji tüketimini belirleyen temel değişkenleri dikkate alır. Bu değişkenler şehirdeki toplam hane sayısı (N), günlük bazda hane başına enerji tüketimi (C_u), mevsimsel faktör ($S(t)$) ve stokastik değişkenlik (ε) oluyor. Günlük enerji tüketimi (E_d) su şekilde hesaplanır;

$$E_d = N(C_u + \varepsilon)S(t)$$

Farz edelim ki $N = 200000$ hane, $C_u = 20 \text{ kWh/gn}$, mevsimsel katsayı yaz için 1.5 ve kış için 1.2 ve günlük enerji tüketimindeki rastgelelik Norm (0,3) olsun. Bu bilgilere göre tablo 1, simülasyondan elde edilen tahminleri ve gerçek enerji tüketim değerlerini göstermektedir.

Gün	Simülasyon sonucu (GWh)	Gerçek değer (GWh)	Fark (%)
Pazartesi	6.2	6.3	1.59
Sali	6.0	6.1	1.64
Çarşamba	6.1	6.4	4.69
Perşembe	6.3	6.2	1.61
Cuma	6.5	6.7	3.08
Cumartesi	7.0	7.3	4.11
Pazar	7.2	7.5	4.00

Tablo 1: simülasyondan elde edilen ve gerçek enerji tüketim değerleri

Tablodaki farklar genellikle %5'in altındadır, bu da modelin doğruluğunun kabul edilebilir olduğunu gösterir. Özellikle modelin genel eğilimleri doğru bir şekilde tahmin ettiği görülmüştür (örneğin hafta sonu tüketimindeki artış). Farkların küçük olduğu durumlar, rastgelelik ε ve model parametrelerinin iyi ayarlandığını gösterir.

2.4 Hassasiyet Analizi (Sensitivity Analysis)

Hassasiyet analizi, bir modelin çıktılarının, girdi parametrelerindeki değişimlere nasıl tepki verdiğini inceleyen bir süreçtir. Modelleme ve simülasyonlarda, özellikle karmaşık ve multidisipliner sistemlerin anlaşılmasında kritik bir rol oynar. Bu yöntem, modelin güvenilirliğini artırmak, kritik girdileri belirlemek ve çıktılar üzerindeki etkilerini değerlendirmek için uygulanır. Ayrıca, sistemlerin hangi girdiler tarafından kontrol edildiğini ve hangi parametrelerin optimize edilmesi gerektiğini anlamaya yardımcı olur. Modelin güçlü ve zayıf yönlerini tespit etmek için hassasiyet analizi temel bir araçtır ve simülasyonların doğruluğu için vazgeçilmezdir.

2.4.1 Hassasiyet Analizinin Tanımı ve Önemi

Hassasiyet analizi, bir modelin, farklı parametre değerlerindeki küçük değişimlere nasıl tepki verdiğini anlamak için kullanılır. Modelleme ve simülasyon süreçlerinde en kritik girdileri belirleyerek sistemin davranışını tahmin etmek için oldukça önemlidir. Bu süreç, parametre optimizasyonu ve hata paylarının azaltılmasında etkili bir araçtır. Özellikle multidisipliner projelerde, farklı disiplinlerden gelen parametrelerin nasıl etkileşimde bulunduğunu anlamak ve bu etkileşimlerin sonuçlarını değerlendirmek için hassasiyet analizi uygulanır.

Hassasiyet analizinin en önemli faydalarından biri, bir modelin hangi girdilere daha duyarlı olduğunu ve hangi girdilerin model çıktıları üzerinde daha fazla etki yarattığını tespit etmektir. Örneğin, bir finansal modelde faiz oranının ekonomik büyümeye etkisini anlamak veya bir biyolojik modelde sıcaklık değişikliklerinin bir organizmanın metabolizmasına etkisini ölçmek gibi konularda kullanılabilir. Doğru girdilerin önceliklendirilmesi, kaynakların daha verimli kullanımına olanak tanır.

2.4.2 Hassasiyet Analizinde Kullanılan Yöntemler

Hassasiyet analizinde iki ana yaklaşım öne çıkar, yerel ve küresel analiz. Yerel hassasiyet analizi, model parametrelerinin belirli bir noktadaki küçük değişimlere duyarlılığını ölçer. Bu analiz, genellikle tek bir parametre çevresinde yapılan basit hesaplamalarla sınırlıdır. Örneğin, bir kimyasal reaksiyon modelinde, sıcaklıktaki küçük bir değişimin

reaksiyon hızına etkisini anlamak için bu yöntem kullanılabilir. Yerel analiz hızlı ve basit bir yöntem olmasına rağmen, tüm parametre uzayını kapsamaması önemli bir dezavantajdır.

Buna karşılık, küresel hassasiyet analizi bir modelin tüm parametre değer aralıklarını dikkate alır ve çıktı üzerindeki etkilerini daha kapsamlı bir şekilde analiz eder. Bu yöntemde genellikle Sobol yöntemi veya Monte Carlo simülasyonları gibi teknikler kullanılır. Örneğin, bir ekonometrik modelde, faiz oranı ve enflasyon gibi faktörlerin ekonomik büyümeye olan etkisi küresel analiz ile değerlendirilebilir. Küresel analiz, daha fazla hesaplama gücü gerektirir ancak çok daha geniş kapsamlı sonuçlar sunar.

2.4.3 Hassasiyet Analizinin Uygulaması

Hassasiyet analizinin modelleme ve simülasyona entegrasyonu, sistemin doğru ve verimli bir şekilde çalışmasını sağlar. Örneğin, bir şehirdeki elektrik tüketimini modelleyen bir sistem ele alındığında, bu modelin toplam enerji tüketimi, sektörel enerji tüketimi ve çalışma süresi gibi girdilere bağlı olduğu görülür. Model şu şekilde ifade edilir;

$$E = \sum_{i=1}^n P_i H_i$$

Bu denklemde P_i i. sektördeki ortalama enerji tüketimini (kWh), H_i ise bu sektördeki toplam çalışma süresini (saat) temsil etmektedir. Hassasiyet analizi, modelin bu girdilere olan duyarlılığını anlamak için yapılır. Yerel hassasiyet analizinde, ticari sektördeki enerji tüketim parametresinde yapılan %10'luk bir artışın toplam enerji tüketimini %5 artırdığı gözlemlenmiştir. Küresel hassasiyet analizi ile Sobol yöntemi kullanılarak, ticari sektörün toplam enerji tüketimindeki varyansın %60'ını oluşturduğu belirlenmiştir. Bu analiz, ticari sektördeki enerji tüketimini optimize etmenin toplam enerji tüketiminde %20'ye kadar tasarruf sağlayabileceğini göstermiştir.

2.4.4 Multidisipliner Uygulama Alanları

Hassasiyet analizi, birçok farklı disiplinde yaygın bir şekilde kullanılmaktadır. İklim biliminde, sera gazı emisyonlarının iklim değişiklikleri üzerindeki etkisi veya sağlık modellemesinde bir aşının

etkinliğini etkileyen biyolojik faktörler bu analizle incelenebilir. Örneğin, bir iklim modelinde karbondioksit seviyeleri, güneş ışıınımı ve okyanus sıcaklığı gibi parametrelerin etkisi hassasiyet analizi ile değerlendirilebilir. Sağlık modellemesinde ise bir aşının bağışıklık oluşturma oranı gibi parametrelerin etkileşimi bu yöntemle analiz edilebilir.

2.5 Karmaşık Sistemler ve Sistem Teorisi

Karmaşık sistemler, farklı bileşenlerin birbirleriyle etkileşime geçtiği, bireysel davranışların sistem düzeyinde beklenmedik sonuçlara yol açabildiği sistemlerdir. Bu sistemler genellikle çok sayıda bileşen içerir, her biri bağımsız olarak hareket eder, ancak toplu etkileşimleri karmaşık bir dinamik ortaya çıkarır. Sistem teorisi ise bu tür sistemlerin analizi, modellenmesi ve anlaşılmasını sağlayan teorik çerçeveyi sunar. Karmaşık sistemler ve sistem teorisi, mühendislik, biyoloji, sosyal bilimler ve ekonomi gibi birçok disiplinde temel bir rol oynar.

Sistem teorisi, karmaşık sistemlerin bileşenleri arasındaki ilişkilere odaklanır. Bu teoriye göre, bir sistemin davranışı, bireysel bileşenlerin toplamından farklıdır. Sistem teorisinin temel bileşenleri şunlardır:

- **Girdiler ve Çıktılar:** Her sistemin bir girdisi ve çıktısı vardır. Örneğin, bir ekosistemde güneş ışığı (girdi), bitki büyümesi (çıkıtı) olarak düşünülebilir.
- **Durum Değişkenleri:** Sistem durumunu belirten parametrelerdir. Örneğin, bir trafik sistemi için araç yoğunluğu bir durum değişkeni olabilir.
- **Geri Besleme Döngüleri:** Sistem davranışlarını düzenler. Örneğin, ekonomik bir sistemde fiyat artışı talebi azaltabilir (negatif geri besleme).
- **Sınırlar:** Bir sistemi çevresinden ayıran unsurlar. Örneğin, bir şehir sınırı, kentsel sistemleri tanımlayabilir.

Aşağıda karmaşık sistemlerin en önemli özellikleri verilmiştir;

- **Hiyerarşik Yapı:** Karmaşık sistemler genellikle alt sistemlerden oluşur. Örneğin, insan vücudu (bir karmaşık sistem) hücreler, organlar ve organ sistemleri gibi alt sistemlerden meydana gelir.

- Öz örgütlenme: Sistem, dış müdahale olmadan kendiliğinden düzenlenebilir. Örneğin, bir karınca kolonisi, her bir karıncanın basit kurallara göre hareket etmesine rağmen karmaşık bir yapı sergiler.
- Geri Besleme Döngüleri: Sistem bileşenleri arasında pozitif veya negatif geri beslemeler vardır. Pozitif geri besleme, sistemin davranışını güçlendirirken; negatif geri besleme, sistemin dengede kalmasını sağlar.
- Adaptif Davranış: Karmaşık sistemler çevresel değişikliklere uyum sağlayabilir. Örneğin, bir ekosistem, iklim değişikliği gibi faktörlere tepki verebilir.
- Bilinmezlik ve Kaos: Karmaşık sistemlerde başlangıç koşullarındaki küçük farklılıklar, büyük sonuçlara yol açabilir. Bu özellik, kaos teorisi ile ilişkilidir.

2.5.1 Karmaşık Sistemlerin Modellenmesi ve Örnekler

Karmaşık sistemlerin modellenmesi, sistemin davranışlarını anlamak ve tahmin etmek için gereklidir. Modelleme sürecinde matematiksel denklemler, hesaplamalı yöntemler veya simülasyonlar kullanılır. Aşağıda karmaşık sistemler için bazı örnekler verilmiştir:

Örnek: Trafik Simülasyonu: Trafik, karmaşık bir sistem olarak düşünülebilir. Her bir araç bağımsız bir bileşen olarak hareket eder, ancak toplu olarak belirli bir trafik dinamiği oluşturur. Örneğin, araçların hızını, yol üzerindeki trafik ışıklarını ve kavşak yoğunluğunu içeren bir model şu şekilde ifade edilebilir:

$$v_i(t + 1) = v_i(t) + a_i(t) - f_i(t)$$

Burada, $v_i(t)$ i numralı aracın hızı, $a_i(t)$ aracın hızlanma miktarı ve $f_i(t)$ frenleme etkisidir.

Bu model, bir otoyol simülasyonunda araç yoğunluğuna göre hız değişimlerini analiz etmek için kullanılabilir.

Örnek: Ekolojik Sistemler: Bir göldeki oksijen seviyesi karmaşık bir sistem olarak modellenenebilir. Balık popülasyonu, alg büyümesi ve su sıcaklığı arasındaki etkileşimler aşağıdaki gibi ifade edilebilir;

$$\frac{dO}{dt} = -k_1 N + k_2 A - k_3 T$$

Burada, O oksijen seviyesi, N balık popülasyonu, A alg yoğunluğu ve T su sıcaklığıdır.

Model, oksijen seviyesinin balık popülasyonu üzerindeki etkisini tahmin edebilir ve bu sistemin dengesini analiz edebilir.

Örnek: Sosyal Ağlar: Sosyal medya platformlarında bilgi yayılımı, karmaşık bir sistemin başka bir örneğidir. Her birey bir düğüm olarak düşünülebilir ve bilgi yayılımı şu şekilde modellenenebilir:

$$I(t + 1) = \alpha I(t)(1 - I(t)) - \beta I(t)$$

Burada, $I(t)$ belirli bir zamanda bilgiye maruz kalan bireylerin oranı, α yayılım oranı ve β bilginin unutulma oranıdır. Bu model, sosyal medyada bir haberin ne kadar yayılacağını öngörmek için kullanılabilir.

2.5.2 Karmaşık Sistemlerin Önemi ve Uygulamaları

Karmaşık sistemler, günümüzün en kritik problemlerini çözmek için hayati bir öneme sahiptir. Örneğin iklim modelleri küresel ısınmayı anlamak için atmosfer, okyanus ve kara etkileşimlerini birleştirir, finansal piyasalarda yatırımcı davranışlarını, piyasa dinamiklerini ve dış etkileri simüle eder ve sağlık sistemlerinde pandemilerin yayılımını modellemek için kullanılır.

Karmaşık sistemlerin anlaşılması, politika yapıcıların daha iyi kararlar almasına, sistemlerin daha verimli hale getirilmesine ve toplumsal sorunların çözülmesine katkı sağlar.

Bölüm 3. Simülasyon Teknikleri

Simülasyon, bir sistemi veya süreci modelleyerek onun davranışını anlamayı, analiz etmeyi ve optimize etmeyi amaçlayan güçlü bir araçtır. Simülasyon teknikleri, farklı sistemlerin dinamiklerini incelemek için çeşitli yaklaşımlar sunar. Bu bölümde ele alınan teknikler, bir sistemin mikro düzeydeki bileşenlerinden makro düzeydeki etkileşimlerine kadar geniş bir perspektifte analiz yapmamıza olanak sağlar.

Bu bölümde, simülasyon teknikleri kapsamlı bir şekilde incelenecek ve her tekniğin temel ilkeleri, kullanım alanları ve avantajları açıklanacaktır. Simülasyon tekniklerinin doğru seçimi, modelleme sürecinin başarısı için kritik bir öneme sahiptir. Her teknik, farklı bir bakış açısı sunduğundan, karmaşık sistemlerin anlaşılmasını kolaylaştırır ve çok disiplinli simülasyon yaklaşımlarında temel yapı taşlarını oluşturur.

Bu bölümde, yedi farklı simülasyon tekniği, ayrık olay simülasyonu, süreç tabanlı simülasyon, sistem dinamiği, monte Carlo simülasyonu, ajan tabanlı simülasyon, kural tabanlı simülasyon ve hibrit simülasyon teknikleri ele alınacaktır.

Her alt bölüm, ilgili tekniğin temel prensiplerini, avantajlarını ve uygulama örneklerini detaylı bir şekilde açıklayacaktır. Amaç, okuyucunun simülasyon teknikleri arasında karşılaştırma yapabilmesi ve uygun durumlarda hangi tekniği kullanacağını kavramasıdır. Simülasyon tekniklerinin her biri, belirli sistem türleri ve problemler için farklı avantajlar sunar. Örneğin ayrık olay simülasyonu, olayların sıralı bir şekilde modellenmesine odaklanırken, sistem dinamiği, sistemin genel dinamiklerini incelemek için kullanılabilir ve ajan tabanlı simülasyon gibi diğer yaklaşımlar ise bireysel bileşenlerin etkileşimleri vurgu yapar.

3.1 Ayrık Olay Simülasyonu (DES)

Ayrık olay simülasyonu (Discrete Event Simulation, DES), sistemlerin zamanla değişen davranışlarını modellemek için kullanılan bir tekniktir. Bu yaklaşım, olayların meydana geldiği belirli zaman noktalarına odaklanır ve her olay, sistemin durumunu değiştirir. DES genellikle lojistik, üretim, sağlık hizmetleri ve telekomünikasyon gibi alanlarda kullanılır.

DES, aşağıdaki temel kavramlara dayanır:

1. Olaylar: Sistem durumunda değişikliğe neden olan belirli eylemler (örneğin, bir müşterinin kasaya gelmesi).
2. Durumlar: Sistemin mevcut hali (örneğin, bir kuyruğun uzunluğu).
3. Süreçler: Olayların sırasını belirleyen mantık.
4. Simülasyon Saati: Sistemin ilerlediği zaman dilimi.

Bu teknik, çeşitli endüstrilerde ve uygulama alanlarında verimliliği artırmak ve süreçleri iyileştirmek için kullanılmaktadır. İşte ayrık olay simülasyonunun yaygın uygulama alanları:

1. Üretim ve Lojistik

Ayrık olay simülasyonu, fabrika üretim hatlarının verimliliğini analiz etmek için etkili bir araçtır. Üretim süreçlerini simüle ederek darboğazları tespit etmek, kaynak kullanımını optimize etmek ve üretim akışlarını iyileştirmek mümkün hale gelir. Bu sayede, üretim maliyetleri azaltılabilir ve teslim süreleri iyileştirilebilir.

2. Sağlık Hizmetleri

Acil servislerde hasta akışını optimize etmek amacıyla ayrık olay simülasyonu kullanılabilir. Simülasyonlar, hasta kabul süreçlerini, doktor ve hemşire dağılımını, bekleme sürelerini ve servis kapasitesini analiz etmeye olanak tanır. Böylece, sağlık hizmetlerinin kalitesi artırılabilir ve hasta memnuniyeti sağlanabilir.

3. Telekomünikasyon

Telekomünikasyon sektöründe, ağ trafiğini ve bant genişliğini simüle etmek için ayrık olay simülasyonu kullanılır. Ağ performansını değerlendirmek, potansiyel tıkanıklıkları önceden belirlemek ve ağ altyapısını planlamak için bu teknikten faydalanılır. Bu sayede, hizmet kalitesi iyileştirilir ve kullanıcı memnuniyeti artırılır.

4. Perakende

Mağaza içi müşteri akışını düzenlemek için ayrık olay simülasyonu kullanılabilir. Müşteri trafiğini analiz ederek satış alanlarının düzenlenmesi, kasa yerleşimi, personel planlaması gibi konularda

iyileştirmeler yapılabilir. Bu, müşteri deneyimini geliştirir ve satışların artmasına katkı sağlar.

Ayrık olay simülasyonu, farklı sektörlerde operasyonel verimliliği artırmak ve süreçleri optimize etmek için vazgeçilmez bir araçtır. Uygulama alanları genişledikçe, bu teknikten elde edilecek faydalar da artmaktadır.

Örnek: Müşteri Kuyruğu Simülasyonu:

Bir markette müşterilerin kasadaki bekleme sürelerini analiz eden bir simülasyon örneği üzerinden DES'i detaylandıralım.

Problemin Tanımı:

Marketin kasasında bir müşteri sırası oluşmaktadır. Amaç, müşterilerin ortalama bekleme süresini hesaplamak ve bu süreyi azaltmak için çözüm önerileri geliştirmektir.

Modelleme Süreci:

Olaylar: Bir müşterinin sıraya girişi, bir müşterinin kasada hizmet alması, hizmet tamamlandıktan sonra müşterinin çıkışı.

Durumlar: Sıradaki müşteri sayısı, kasiyerin meşgul ya da boşta olması.

Girdi Parametreleri: Müşterilerin sıraya giriş sıklığı (örneğin, her 2 dakikada bir), hizmet süresi (örneğin, müşteri başına 3 dakika).

Müşteri bekleme süresi şu şekilde hesaplanabilir:

$$B = \alpha \cdot \frac{E(S^2)}{2(1 - \beta)}$$

Burada, B ortalama bekleme süresi, α müşteri giriş oranı, $E(S^2)$ hizmet süresinin ikinci momenti ve $\beta = \frac{\alpha}{\gamma}$ (γ hizmet oranı) sistem yoğunluğudur.

Zaman ilerleyişi ve olayların sırasını anlamak için bir tablo oluşturulabilir:

Zaman (dk)	Olay	Sıradaki müşteri sayısı	Kasiyer durumu	Bekleme süresi
0	Müşteri giriş yaptı	1	Meşgul	0
2	Yeni müşteri geldi	2	Meşgul	2
3	Hizmet tamamlandı	1	Meşgul	0
4	Yeni müşteri geldi	2	Meşgul	1

Tablo 2: Zaman ilerleyişi

Simülasyon sonuçlarına göre, bekleme süresini azaltmak için kasiyer sayısını artırma veya hizmet süresini kısaltma gibi öneriler geliştirilebilir.

3.2 Süreç Tabanlı Simülasyon (Process-Based Simulation)

Süreç tabanlı simülasyon (PBS), bir sistemin işleyişini adım adım ele alan ve süreçlerin sırasına odaklanan bir simülasyon türüdür. Bu yaklaşım, özellikle karmaşık süreçlerin detaylı analizini yapmak için kullanılır. Sistem içerisindeki her süreç, belirli bir sıra ve mantık çerçevesinde ilerler. PBS, ayrıık olay simülasyonuna benzer bir mantık taşır ancak süreçlere ve akışa daha fazla vurgu yapar.

Süreç tabanlı simülasyon aşağıdaki temel bileşenlere dayanır:

1. Süreçler: Sistemde gerçekleşen ardışık işlemleri tanımlar.
2. Kaynaklar: Süreçleri gerçekleştirmek için gereken varlıklar (örneğin, makineler, insanlar).
3. Süreler: Süreçlerin tamamlanması için geçen zaman.
4. Olaylar: Bir sürecin başlangıcını veya sonunu belirleyen tetikleyiciler.

PBS genellikle tedarik zinciri yönetimi, üretim süreçleri, sağlık hizmetleri ve lojistik gibi alanlarda tercih edilir.

Örnek 1: Üretim Hattı Simülasyonu

Bir fabrika, üretim hattındaki verimliliği artırmak için simülasyon yapmayı planlıyor. Hedef, üretim hattındaki darboğazları tespit etmek,

bekleme sürelerini azaltmak ve üretim kapasitesini artırmak. Üretim hattı üç ana aşamadan oluşuyor:

1. Malzeme Hazırlama: Ham maddelerin işleme hazır hale getirilmesi.
2. Montaj: Parçaların birleştirilip ürün oluşturulması.
3. Kalite Kontrol: Üretilen ürünlerin kontrol edilip sevkiyata gönderilmesi.

Her aşama için işlem süreleri ve kaynak bilgileri aşağıdaki gibidir:

Süreç	İşlem Süresi (dk)	Kaynak Sayısı
Malzeme Hazırlama	5	2
Montaj	15	3
Kalite Kontrol	10	1

Tablo 3: Süreç, İşlem Süresi ve Kaynak Sayısı

Tablo 3 malzeme hazırlama süreçte giriş yapan ürünler 2 iş istasyonu tarafından, montaj süreçte ürünler bir sonraki aşamaya aktarılır ve 3 montaj istasyonu tarafından ve kalite kontrol aşamada ürünler kalite kontrolünden geçirilir ve sevkiyata hazır hale geldiğini gösteriyor.

Simülasyonda üretim hattına giriş oranı dakikada bir ürün, İşlem süreleri normal dağılımı (standart sapma ± 2 dakika) ve günlük çalışma süresi 8 saat (480 dakika) olarak tanımlanır.

Her işlem için hizmet oranı $\mu = \frac{1}{T_{service}}$ olarak hesaplanır. Örneğin, malzeme hazırlama sürecinde hizmet oranı $\mu = \frac{1}{5} = 0.2$ oluyor.

M/M/c (Markovian varış süreci, Markovian hizmet süreci, çoklu sunucu sayısı) kuyruğu kullanılarak kuyruk uzunluğu ve bekleme süreleri hesaplanabilir:

$$L_q = \frac{\left(\frac{\lambda}{\mu}\right)^c \left(\frac{\lambda}{\mu}\right)}{c! \left(1 - \left(\frac{\lambda}{c\mu}\right)\right)^2} \quad W_q = \frac{L_q}{\lambda}$$

Burada λ giriş oranı (ürün/dk), c kaynak sayısı ve μ hizmet oranıdır. Bu hipotezlere göre zaman akışı aşağıdaki tabloda verilmiştir.

Zaman (dk)	Süreç	Malzeme Hazırlama	Montaj	Kalite Kontrol	Kuyrukta Bekleyen Ürün	Sevk Edilen Ürün
0	Ürün 1 giriş yaptı	1	0	0	0	0
5	Ürün 1 montajda	0	1	0	0	0
20	Ürün 1 kalite kontrolde	0	0	1	0	0
30	Ürün 1 sevk edildi	0	0	0	0	1

Tablo 4: Zaman akışı

Simülasyon sonuçlarına göre en fazla bekleme süresi montaj sürecinde gözlenmiştir. Bunun nedeni, montaj aşamasındaki işlem süresinin uzunluğu ve kaynak sayısının yetersizliğidir. Çözüm olarak montaj aşamasında bir ek kaynak sağlanması, üretim kapasitesini %20 artırabilir.

Verimlilik analizi aşamasında mevcut kaynaklarla günlük maksimum üretim kapasitesi $Kapasite = \frac{\text{Toplam Alma Süresi}}{\text{ortalama süre}}$ formüle ile hesaplanabilir. Örneğin, montaj süreci için kapasite $\frac{480}{15} = 32$ olarak hesaplanır. Sonuç olarak kalite kontrol sürecinde otomasyon, işlem süresini 5 dakikaya indirebilir.

Örnek 2: Hastane Polikliniğinde Hasta Akışı Simülasyonu

Bir hastane polikliniği, günlük hasta akışını optimize etmek istiyor. Amaç, hastaların bekleme süresini azaltmak ve sağlık personelinin iş yükünü dengelemek. Poliklinikte dört temel süreç vardır:

1. Kayıt: Hastaların giriş yapıp bilgilerinin alınması. Ortalama işlem süresi 3 dakika.

2. Muayene: Doktor tarafından hastanın muayene edilmesi. Ortalama işlem süresi 10 dakika.

3. Laboratuvar: Gerekirse testlerin yapılması. Ortalama işlem süresi 15 dakika.

4. Çıkış İşlemleri: Hastanın ödeme yapıp poliklinikten ayrılması. Ortalama işlem süresi 5 dakika.

Bu hastanede kaynak olarak kayıt için 1 personel, muayene için 2 doktor, laboratuvar için 1 teknisyen ve çıkış için 1 personel görevlendirilmiştir. Hastaneye her 5 dakikada bir yeni hasta gelir ve süreç eksponansiyel dağılımıdır. Bu hipotezlere göre zaman akışı aşağıdaki tablada verilmiştir.

Zaman (dk)	Olay	Kayıt ki Hasta	Muayene eki Hasta	Laboratuvar aki Hasta	Çıkış ki Hasta	Tamamlan an Hasta
0	İlk hasta kayıtta	1	0	0	0	0
3	İlk hasta muayenede	0	1	0	0	0
13	İlk hasta laboratuvar da	0	0	1	0	0
28	İlk hasta çıkışta	0	0	0	1	0
33	Yeni hasta kayıtt	1	0	0	0	1

Tablo 5: Zaman akışı

Süreçlerin toplam bekleme süresi ve servis süresi aşağıdaki formüllerle hesaplanabilir:

$$\text{Bekleme Süresi } W = T_{servis} + T_{bekleme}$$

Burada W Toplam süreçte geçirilen zaman, T_{servis} servis süresi (kayıt, muayene, laboratuvar, çıkış) ve $T_{bekleme}$ süreç öncesi bekleme süresidir.

Her bir süreçteki hizmet oranı $\mu = \frac{1}{T_{service}}$ olarak hesaplanır. Varsayılan simülasyonda analiz her süreç için kuyruk uzunluğu ve ortalama bekleme süreleri aşağıdaki gibi yapılır;

$$L_q = \frac{\lambda^2}{\mu(\mu - \lambda)} \quad W_q = \frac{L_q}{\lambda}$$

Burada λ hasta giriş oranı ve μ servis oranıdır.

Simülasyon sonuçlarına göre en fazla bekleme süresi laboratuvar sürecinde gerçekleşmiştir. Bunun nedeni, laboratuvarda yalnızca bir teknisyenin çalışmasıdır. Laboratuvarda ikinci bir teknisyen çalıştırmak, bekleme süresini %40 oranında azaltabilir. Sonuç olarak, süreç tabanlı

simülasyon, poliklinik süreçlerini optimize etmek ve kaynak kullanımını dengelemek için güçlü bir yöntem sunar. Bu analizle, hastaların bekleme süresi azaltılmış ve sistem genelindeki kaynakların verimliliği artırılmıştır.

Bu iki örnekte de süreç tabanlı simülasyon tercih edilmesinin nedeni, sistemlerin her bir işlem adımını modellemek ve kaynak kullanımı ile kuyruk davranışını analiz etmektir. Ancak diğer simülasyon tekniklerinden farklı olarak süreç tabanlı yaklaşımlar, daha çok belirli bir sırayla çalışan ve belirgin işlem sürelerine sahip sistemlerde kullanılır.

3.3 Sistem Dinamiği

Sistem dinamiği, karmaşık sistemlerin zaman içindeki davranışlarını anlamak ve analiz etmek için kullanılan bir modelleme ve simülasyon yaklaşımıdır. Bu yaklaşım, sistem içindeki geri besleme döngüleri, stoklar, akışlar ve gecikmeler gibi unsurları dikkate alarak, sistemin dinamik yapısını ve davranışını incelemeyi sağlar. Sistem dinamiği, mühendislik, ekonomi, ekoloji, sosyoloji ve yönetim bilimleri gibi çeşitli disiplinlerde uygulanmaktadır.

3.3.1 Temel Kavramlar ve Modelleme Süreçleri

Bu bölümde sistem dinamiğinin temel ilkelerini ve modelleme süreçlerini ele alacağız. Sistem dinamiğin temel kavramları aşağıdaki gibidir.

1. Stoklar (Depolar): Sistemde biriken ve zamanla değişen kaynaklardır. Örneğin, bir su rezervuarındaki su miktarı veya bir şirketin envanteri stoklara örnek olarak verilebilir.

2. Akışlar: Stokların giriş ve çıkış hızlarını temsil eder. Örneğin, su rezervuarına akan su miktarı giriş akışını, rezervuardan çıkan su miktarı ise çıkış akışını temsil eder.

3. Geri Besleme Döngüleri: Sistemin çıktılarının tekrar sisteme giriş olarak etki etmesi durumudur. Pozitif geri besleme döngüleri sistemi büyütme veya hızlandırma eğilimindeyken, negatif geri besleme döngüleri dengeleyici veya yavaşlatıcı etki yapar.

4. Gecikmeler: Sistemdeki bir deęişiklięin etkisinin ortaya çıkmasındaki zaman farkıdır. Gecikmeler, sistemin dinamik davranışını önemli ölçüde etkileyebilir.

Ve sistem dinamięinin modelleme süreci 5 adımla tanımlanır;

1. Problemin Tanımı: İlk adım, analiz edilecek sistemin sınırlarını belirlemek ve temel sorunu tanımlamaktır.

2. Sistem Yapısının Belirlenmesi: Stoklar, akışlar, geri besleme döngüleri ve gecikmeler gibi temel bileşenler tanımlanır ve aralarındaki ilişkiler belirlenir.

3. Matematiksel Modelleme: Sistemin davranışını tanımlayan diferansiyel denklemler oluşturulur.

4. Simülasyon: Matematiksel model, bilgisayar ortamında simüle edilerek sistemin zaman içindeki davranışı analiz edilir.

5. Politika Testleri ve Analiz: Farklı senaryolar ve politikalar simülasyon modeli üzerinde test edilerek, sistemin nasıl tepki vereceęi incelenir.

Bu bölümün devamında gerçekçi bir örnek üzerinden bir sistem dinamięi modeli kurarak analiz yapacağız.

Örnek: Bir Şehirde Su Kaynaklarının Yönetimi

Bir şehirde su tüketimi giderek artmakta ve rezervuardaki su seviyesi düşmektedir. Şehir yönetimi, mevsimsel yağış deęişimlerini, nüfus artışını ve geri dönüşüm oranlarını dikkate alarak su kaynaklarının sürdürülebilir yönetimini sağlamak istemektedir. Sistem dinamięi simülasyon kullanarak bu sorulara yanıt arayacağız:

1. Mevsimsel yağış deęişimleri su seviyesi üzerinde nasıl bir etki yapar?
2. Nüfusun artışı uzun vadede su rezervine nasıl etki eder?
3. Geri dönüşüm oranının artırılması su yönetimini nasıl etkiler?
4. Belirli bir politika uygulanmazsa rezervuarın kuruma riski var mı?

Bu senaryoya uygun stok-akış diyagramı ve diferansiyel denklemlerle tanımlı bir model kuracağız. Sistemin temel stok-akış birinci mertebeden adi diferansiyel denklemi aşağıdaki gibidir;

$$\frac{dS}{dt} = (I(t) + R(t)) - (C(t) + L(t))$$

Burada $S(t)$ rezervuardaki su miktarı (m^3), $I(t)$ rezervuara giren su (yağış ve geri dönüşüm) ($m^3/gün$), $C(t)$ günlük su tüketimi ($m^3/gün$), $L(t)$ buharlaşma ve sızıntı kayıpları ($m^3/gün$) ve $R(t)$ Geri dönüşüm oranına bağlı olarak kazanılan su ($m^3/gün$) olarak tanımlanır.

Yağışın yıllık döngüsel değişimi trigonometrik bir fonksiyon ile modellenir:

$$I(t) = I_0 + A \sin(\omega t)$$

Burada I_0 ortalama yağış miktarı, A yağış değişim genliği ve ω yıllık döngü frekansdır.

Şehir nüfusu lojistik büyüme modeli ile ifade edilir:

$$\frac{dN}{dt} = rN(1 - \frac{N}{N_{maks}})$$

Burada $N(t)$ şehir nüfusu (kişi), r nüfus artış oranı ve N_{maks} şehirdeki maksimum taşıma kapasitesidir.

Su tüketimi nüfus ile doğru orantılıdır:

$$C(t) = c_p N(t)$$

Burada c_p kişi başına günlük su tüketimini ifade ediyor.

Geri dönüşüm oranı artırılırsa, tüketilen suyun bir kısmı tekrar sisteme kazandırılacaktır:

$$R(t) = k_r C(t)$$

Burada k_r Geri dönüşüm oranıdır.

Eğer k_r artırılırsa, sistemin uzun vadeli dengesine etkisini analiz edebiliriz.

Buharlaşma ve sızıntılar sıcaklığa bağlı olarak periyodik değişebilir ve değişim aşağıdaki denklem ile analiz edilebilir;

$$L(t) = L_0 + B \cos(\omega t)$$

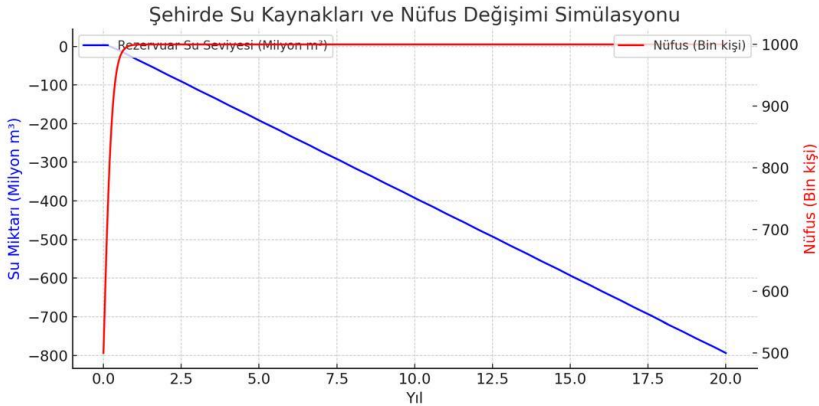
Burada L_0 Ortalama kayıp miktarı ve B Mevsimsel değişimi göstermektedir.

Bu adımda, denklemlerden sonra, başlangıç değer problemi tanımlanarak simülasyon yapıp ve analiz edebiliriz. Simülasyon parametreleri için başlangıç değerleri aşağıdaki tabelada verilmiştir.

Parametre	Değer	Açıklama
I_0	20000 m ³ /gün	Ortalama yağış
A	5000 m ³ /gün	Yağış değişim genliği
ω	$2\pi/365$	Yıllık yağış döngüsü
c_p	150 L/kişi/gün	Kişi başına su tüketimi
k_r	0.2	Geri dönüşüm oranı
N_0	500000 kişi	Başlangıç nüfusu
r	0.02	Nüfus büyüme oranı
N_{maks}	1000000 kişi	Maksimum nüfus
L_0	10000 m ³ /gün	Ortalama su kaybı
B	2000 m ³ /gün	Mevsimsel değişim genliği

Tablo 6: Parametrelerin Başlangıç Değerleri

Bu değerlere göre simülasyon sonucunu görmek için modeli kodlayarak verileri analiz edebiliriz. Bunun için, diferansiyel denklemleri çözüp rezervuar su seviyesi, nüfus değişimi ve su tüketimi gibi parametreleri zaman içinde nasıl değiştiğini grafiğe dökebiliriz.



Grafik 1: Şehirdeki Rezervuar Su Seviyesi ve Nüfus Değişimini

Grafik 1, 20 yıllık süre boyunca şehirdeki rezervuar su seviyesi (mavi) ve nüfus değişimini (kırmızı) göstermektedir. Su seviyesi zamanla azalma eğilimindedir, ancak mevsimsel yağış değişimleri nedeniyle dalgalanmalar görülmektedir. Nüfus büyümesi başlangıçta hızlanırken, sınıra yaklaştıkça yavaşlamaktadır (lojistik büyüme modeli). Artan nüfus nedeniyle su tüketimi de artmaktadır, bu da rezervuar seviyesini düşüren ana etkenlerden biridir.

Bu sonuçlarla başlangıçta sorulan soruların cevabını verebiliriz. Birinci ve ikinci sorunun cevabı yukarıda verilmiştir. Şimdi geri dönüşüm oranı atılırsa ne olur sorunun analizi için geri dönüşüm oranı $k_r = 0.2$ yerine $k_r = 0.4$ alıyoruz ve simülasyonu bir daha bu değer ile işleme alıyoruz. Bu durumda görüyoruz ki su kaybı azalıyor ve rezervuar seviyesi daha kararlı hale geliyor ve kuruma riski düşüyor. Sonuç olarak, geri dönüşüm sistemine yapılan yatırımlar, uzun vadede su kaynaklarının korunmasını sağlar. Bu analizlere göre politikalar uygulanmazsa nüfus 800.000'i aştığında su talebi, yağış ve geri dönüşümle karşılanamaz hale geliyor ve eğer geri dönüşüm yapılmazsa rezervuarın 20 yıl içinde kritik seviyeye düşeceği görülmüyor. Sonuç olarak şehir yönetimi geri dönüşüm oranını artırmalı ve nüfus artışını dikkate alarak yeni su kaynakları oluşturmmalıdır.

Sistem dinamiği simülasyonu baz alarak aşağıdaki politikalar önerilmektedir.

- Geri dönüşüm sistemleri güçlendirilmeli: Geri dönüşüm oranı %40'a çıkarıldığında rezervuar seviyesi daha dengeli kalmaktadır.
- Su tüketimi bilinçlendirme kampanyaları ile azaltılmalı: Kişi başına su tüketimi düşürülerek talep yönetilebilir.
- Yeni su kaynakları oluşturulmalı: Alternatif su kaynakları (yer altı suyu, deniz suyu arıtma) incelenmelidir.

Bu analizde sistem dinamiği yöntemi ile su kaynaklarının yönetimi incelenmiştir. Gerçek verilere dayalı bir simülasyon ile uzun vadeli politikalar test edilebilir.

3.4 Monte Carlo Simülasyon (MCS)

Monte Carlo Simülasyonu, stokastik süreçlerin modellenmesi için kullanılan güçlü bir matematiksel yaklaşımdır. Olasılığa dayalı sistemlerin analizinde, çok sayıda rastgele örnekleme yoluyla sonuçların dağılımını tahmin etmeye yarar. Finans, mühendislik, fizik, ekonomi ve lojistik gibi birçok alanda karar destek mekanizmalarında kritik bir rol oynar.

Monte Carlo simülasyonunun temel süreci şu adımlardan oluşur:

1. Problemin Tanımlanması: Analiz edilmek istenen sistemin girdileri, değişkenleri ve belirsizlik unsurları belirlenir.
2. Olasılık Dağılımlarının Belirlenmesi: Rastgele değişkenler için uygun olasılık dağılımları atanır (örneğin, normal, üstel, log normal dağılımlar).
3. Rastgele Sayı Üretimi: Simülasyonun yürütülmesi için belirlenen dağılımlardan rastgele sayılar üretilir.
4. Simülasyonun Çalıştırılması: Belirlenen model, yüzlerce veya milyonlarca kez rastgele girdilerle çalıştırılır.
5. Sonuçların Analizi: Çıktılar istatistiksel yöntemlerle analiz edilerek olasılık dağılımları ve risk değerlendirmesi yapılır.

Monte Carlo simülasyonu, olasılık teorisi ve istatistik kullanılarak modellenir. Örneğin, bir varlığın yıllık getirisi R , normal dağılıma sahip bir rastgele değişken olarak modellenebilir:

$$R \sim N(\mu, \sigma^2)$$

Burada, μ beklenen getiri ve σ standart sapma veya risktir. N dönemlik bileşik getiri şu şekilde hesaplanır:

$$P_T = P_0 * \prod_{t=1}^T (1 + R_t)$$

Burada P_T , T yıl sonundaki portföy değeri; P_0 başlangıç yatırımı ve R_t her yılın rastgele getirisidir.

Örnek: Portföy Getirisinin Simülasyonu

Bir yatırımcı, üç farklı varlığa (A, B ve C) yatırım yapmaktadır. Portföy dağılımı 40% A, 30% B ve 30% C gibidir. Varlıkların beklenen yıllık getirileri ve standart sapmaları aşağıdaki gibidir.

Varlık	Beklenen Getiri μ	Standart Sapma σ
A	8%	15%
B	6%	10%
C	4%	7%

Tablo 7: Varlıkların Beklenen Getiri ve Standart Sapması

Varlık getirileri normal dağılıma sahiptir. Yani;

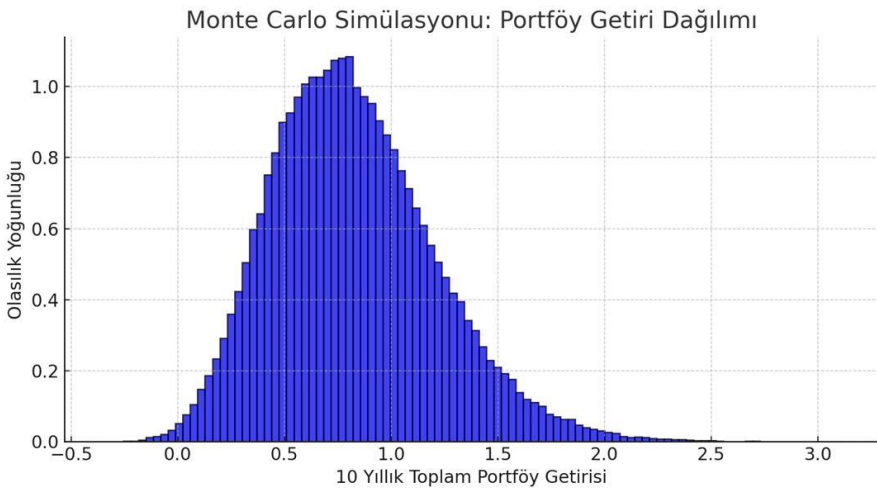
$$r_A \sim N(0.08, 0.15)$$

$$r_B \sim N(0.06, 0.10)$$

$$r_C \sim N(0.04, 0.07)$$

Monte Carlo simülasyonda, 100.000 iterasyon çalıştırılarak 10 yıl sonunda portföy getirisinin dağılımı analiz edelim. Histogram grafiği, portföyün 10 yıl sonunda elde edebileceği getirilerin dağılımını göstermektedir. Bu analiz, yatırımcının potansiyel risk ve getirilerini değerlendirmesine yardımcı olur.

Monte Carlo simülasyonunun sonuçlarına göre beklenen 10 yıllık toplam getiri yaklaşık %82.4 ve risk (standart sapma) yaklaşık %38.6 belirlenmiştir.



Grafik 2: Portföy Getiri Dağılımı

Bu örnek, Monte Carlo simülasyonunun belirsizlik içeren sistemlerde nasıl kullanılabileceğini göstermektedir. Modelin doğruluğu artırılmak istenirse, simülasyon sayısı artırılabilir veya farklı olasılık dağılımları kullanılabilir.

3.5 Ajan-Tabanlı Simülasyon (ABM: Agent Based Modeling)

Ajan Tabanlı Simülasyon (ABM), sistemdeki bireysel bileşenlerin (ajanların) bağımsız kararlar alarak ve çevreleriyle etkileşimde bulunarak sistemin genel dinamiklerini nasıl oluşturduğunu anlamaya yönelik bir yaklaşımdır. Karmaşık adaptif sistemleri modellemek için güçlü bir yöntemdir ve sosyal bilimler, biyoloji, ekonomi, lojistik ve mühendislik gibi birçok alanda kullanılmaktadır.

3.5.1 Ajan Tabanlı Simülasyonun Temel Unsurları

1. Ajanlar (Agents):

- Sistemdeki bireysel bileşenlerdir.
- Özerktirler, yani kendi başlarına karar verebilirler.
- Çevrelerini algılayabilir ve buna göre hareket edebilirler.
- Basit kurallara sahiptirler, ancak büyük ölçekli karmaşık davranışlar ortaya çıkarabilirler.

2. Ortam (Environment):

- Ajanların içinde hareket ettiği ve etkileşime girdiği simülasyon alanıdır.
- Fiziksel bir harita, sosyal bir ağ veya soyut bir sistem olabilir.

3. Etkileşim Kuralları (Interaction Rules):

- Ajanların birbirleriyle ve ortamla nasıl etkileşim kuracaklarını belirler.
- Oyun teorisi, karar ağacı, olasılıksal kurallar veya makine öğrenmesi teknikleri içerebilir.

4. Adaptasyon (Adaptation):

- Ajanlar deneyimlerinden öğrenebilir ve davranışlarını değiştirebilir.
- Genetik algoritmalar veya pekiştirmeli öğrenme gibi teknikler kullanılabilir.

3.5.2 Ajan Tabanlı Simülasyonun Avantajları

Ajan tabanlı simülasyon (ATS), bireysel ajanların etkileşimleri ve davranışlarına dayanarak sistemlerin dinamiklerini analiz etme yöntemidir. Bu simülasyon türü, özellikle karmaşık sistemlerin anlaşılmasında ve modellenmesinde önemli avantajlar sunmaktadır.

1. Gerçekçi ve Mikro-Düzey Analiz

- Bireysel Kararların Makro Ölçekteki Etkileri: ATS, bireylerin aldığı kararların toplamda büyük ölçekli sonuçlar doğurabileceğini gösterir. Bu, bireysel düzeydeki değişimlerin toplumsal veya ekonomik sistemlere nasıl etki ettiğini anlamak için kritik öneme sahiptir.
- Dinamik Model Oluşturma: Karar alma süreçlerini temel alarak, sistemlerin zaman içindeki değişimini ve gelişimini gerçekçi bir şekilde simüle eder.

2. Heterojen Bireyleri Modelleme Yeteneği

- Bireysel Farklılıklar: Klasik diferansiyel denklemlerin aksine, ATS her ajanın farklı kurallar ve özelliklere sahip olmasını sağlar. Bu, sistemdeki her bireyin özgün davranışlarını ve kararlarını yansıtabilir.

3. Olasılıksal ve Stokastik Süreçleri Dahil Eder

- Rastgelelik ve Belirsizlik: ATS, rastgelelik ve belirsizlik içeren sistemlerin modellenmesine olanak tanır. Bu, gerçek dünyadaki belirsizlikleri ve değişkenlikleri daha iyi temsil eder.

4. Adaptif Davranışları Modeller

- Öğrenme ve Karar Mekanizmaları: Ajanlar, çevrelerinden öğrenebilir ve stratejilerini zaman içinde değiştirebilir. Bu, sistemin adaptif dinamiklerini anlamak için önemlidir.

Uygulama Alanları

Ajan tabanlı simülasyonlar geniş bir uygulama yelpazesine sahiptir:

- Salgın Hastalık Modelleme: COVID-19 gibi hastalıkların yayılımını analiz etmek için kullanılır.
- Trafik Akışı Simülasyonu: Şehir içi trafik yönetimini optimize etmek için faydalıdır.
- Ekonomik Piyasa Analizi: Bireysel yatırımcı kararlarının piyasa üzerindeki etkilerini incelemek için kullanılır.
- Lojistik ve Tedarik Zinciri Yönetimi: Stok akışları ve dağıtım optimizasyonu gibi konularda çözüm sunar.
- Sosyal Dinamikler ve Grup Davranışları: Sosyal medya etkileşimleri gibi sosyal sistemlerin modellenmesine olanak tanır.

Bu modellemeler sayesinde, politikalar geliştirilebilir, risk analizleri yapılabilir ve sistemlerin daha iyi anlaşılması sağlanabilir. Ajan tabanlı simülasyonlar, karmaşık sistemlerin daha etkin ve verimli bir şekilde yönetilmesine katkı sağlar

3.5.3 Matematiksel Modelleme ve Simülasyon Süreci

Ajan tabanlı simülasyonlar genellikle stokastik süreçler içerdiği için olasılık teorisi ve karar kuralları ile modellenir. Aşağıda genel bir ajan modeli verilmiştir:

Bir ajan i için durumu $S_i(t) = \{x_i, y_i, v_i, d_i\}$ ile veriliyor. Burada x_i ve y_i ajanın koordinatları $\begin{cases} x_i(t+1) = x_i(t) + v_i \cos(d_i) \\ y_i(t+1) = y_i(t) + v_i \sin(d_i) \end{cases}$, v_i ajanın hızı ve d_i ajanın yönüdür. Ajanın yönü belirli bir olasılık dağılımına göre değişebilir.

$$d_i \sim U(0, 2\pi)$$

Ajanlar birbirine belirli bir mesafe içinde yaklaştığında etkileşim gerçekleşir. Örneğin, $P(A \rightarrow B) = \frac{1}{1 + e^{-x_A + x_B}}$ ajan A'nın ajan B ile etkileşime girme olasılığını gösterir.

Örnek: Salgın Hastalık Yayılımının Simülasyonu

Bir şehirde bulaşıcı bir hastalık yayılmaktadır. İnsanlar belirli hızlarla rastgele hareket etmekte ve birbirleriyle temas ettiklerinde hastalığı yayabilmektedir. Ajan tabanlı simülasyon ile hastalığın yayılım dinamikleri analiz edilmek istenmektedir.

Simüle edilen sistem SIR (Duyarlı-Enfekte-İyileşmiş) modeli temel alınarak oluşturulmuştur. Sistemde N ajanı rastgele bir ortamda hareket etmekte ve temas ettiklerinde belirli bir olasılıkla hastalığı bulaştırmaktadır. Hastalığın yayılım dinamikleri SIR modeli ile aşağıdaki diferansiyel denklemlerle tanımlanır:

$$\frac{dS}{dt} = -\beta SI$$

$$\frac{dI}{dt} = \beta SI - \gamma I$$

Ve

$$\frac{dR}{dt} = \gamma I$$

Burada $S(t)$ duyarlı birey sayısı, $I(t)$ enfekte birey sayısı, $R(t)$ İyileşmiş birey sayısı, β enfeksiyon katsayısı (bulaşma oranı) ve γ İyileşme oranıdır (hastalığın süresi).

Bu diferansiyel denklemler, ajan tabanlı modelde bireysel ajanların etkileşimleriyle belirlenmektedir. Simülasyon Kurulumu için başlangıç koşulları aşağıda verilmiştir.

$N = 1000$ birey rastgele hareket etmektedir.

Başlangıçta %1 oranında birey enfekte (10 kişi).

Her birey belirli hızda rastgele yürümektedir.

Bulaşma mesafesi: 2 metre içinde temas eden bireyler hastalığı kapabilir.

Bulaşma olasılığı: $P_{inf} = 0.1$ (yani %10 temas anında bulaşma şansı).

İyileşme süresi: 7 gün (Poisson dağılımı kullanılarak belirlenmiştir).

Simülasyon sonucunda hastalığın yayılım eğrisi aşağıda verilmiştir. Burada ajan tabanlı modelin simülasyon çıktısı ile diferansiyel denklem tabanlı teorik SIR modeli karşılaştırılmıştır.

Zaman (Gün)	Duyarlı (S)	Enfekte (I)	İyileşmiş (R)
0	990	10	0
5	850	120	30
10	500	400	100
15	200	500	300
20	100	200	700
25	50	50	900
30	10	10	980

Tablo 8: Hastalığın Yayılım Eğrisi

Tablo 8, hastalığın 15. gün civarında zirveye ulaştığını ve zamanla iyileşen birey sayısının arttığını göstermektedir.

Simülasyon sonuçlara göre bulaşma oranı $\beta = 0.1$ ve iyileşme oranı $\gamma = \frac{1}{7}$ için enfekte birey sayısı yaklaşık 15. günde maksimuma ulaşmıştır ve enfekte birey oranı yaklaşık %50'ye ulaşmıştır.

R_0 (Temel Üreme Sayısı), bir enfekte bireyin kaç kişiye hastalığı bulaştırdığını gösteren önemli bir parametredir ve şu formülle hesaplanır:

$$R_0 = \frac{\beta}{\gamma}$$

Burada $\beta = 0.1$ ve $\gamma = \frac{1}{7}$ olduğundan dolayı, $R_0 = 0.7$, analize göre eğer $R_0 > 1$ ise salgın yayılacaktır, eğer $R_0 < 1$ ise salgın zamanla sönmülenecektir. Bu simülasyonda hastalığın kontrolsüz yayılması durumunda kritik bir seviyeye ulaşabileceği görülmektedir. Kontrol ve önlemler için sosyal mesafe artırılırsa bulaşma katsayısı β düşer ve salgın daha yavaş yayılır. Aşı veya bağışıklık kazanımı artırılırsa duyarlı birey sayısı azalır ve hastalık kontrol altına alınır.

Aşağıdaki Python kodu ile yapılan grafik, ajan tabanlı simülasyon yöntemiyle enfekte birey sayısının zaman içindeki değişimini görselleştirmektedir.



Grafik 3: Enfekte Birey Sayısının Zaman İçindeki Değişimi

Bu grafiğe göre, ilk günlerde enfekte birey sayısı düşük seviyededir (10 kişi) ve hastalığın bulaşma süreci yeni başladığı için vaka sayısı yavaşça artmaktadır. Sonrasında, 5.günden itibaren enfekte birey sayısında hızlı bir yükseliş gözlemlenmektedir ve 10.gün civarında enfekte birey sayısı 400’e ulaşarak salgının büyüdüğünü göstermektedir. 15.gün zirve noktasıdır; enfekte birey sayısı maksimum seviyeye (500 kişi) ulaşmıştır. Ve bu noktada düşüş dönemi başlıyor. 15.günden sonra enfekte birey sayısı azalmaya başlamaktadır. Bunun temel nedenleri bağışıklık kazanan veya iyileşen birey sayısının artması, yeni enfekte olabilecek duyarlı bireylerin azalması ve Önlemler veya müdahaleler (örneğin, sosyal mesafe, karantina) salgının yayılım hızını düşürmektedir. Gördüğümüz üzere, 30.gün itibarıyla enfekte birey sayısı tekrar başlangıç seviyesine (10 kişi) gerilemiştir.

Bu analiz ile ajan tabanlı simülasyonun salgın hastalıkların yayılmasını modellemede nasıl kullanılabileceğini detaylı olarak incelemiş olduk. Simülasyonun sonuçları, halk sağlığı politikalarının oluşturulmasına ve salgın kontrol önlemlerinin planlanmasına katkı sağlayabilir. Ayrıca, salgının zirve noktasını belirlemek, sağlık sistemlerinin kapasitesini yönetmek açısından kritik öneme sahiptir.

3.6 Kural tabanlı simülasyon

Kural tabanlı simülasyon, sistemdeki bileşenlerin belirli kurallara göre etkileşime girdiği ve bu kurallar doğrultusunda sistemin zamanla nasıl evrildiğini modelleyen bir tekniktir. Bu yaklaşım, özellikle açıkça tanımlanmış kuralların olduğu sistemlerde oldukça etkilidir. Örneğin, biyolojik süreçlerin modellenmesi, üretim hatlarının simülasyonu veya trafik akış analizleri gibi alanlarda yaygın olarak kullanılır.

Bu tür simülasyonlar, karar mekanizmalarının matematiksel formüller yerine kurallar dizisiyle tanımlandığı sistemlerde avantaj sağlar. Kural tabanlı simülasyonların en büyük faydası, karmaşık sistemleri sezgisel ve modüler bir şekilde tanımlamaya olanak tanımasıdır.

3.6.1 Kural Tabanlı Simülasyonun Temelleri

Kural tabanlı simülasyonun temel bileşenleri şunlardır:

- **Durum (State):** Sistemin belirli bir andaki mevcut durumu.
- **Kurallar (Rules):** Belirli koşullar sağlandığında sistemin nasıl değişeceğini belirleyen karar mekanizmaları.
- **Ajanlar (Agents) veya Birimler:** Kurallara uygun şekilde hareket eden ve etkileşime giren bileşenler.
- **Geri Bildirim Mekanizması:** Sistemin belirli durumlara nasıl tepki verdiğini düzenleyen süreçler.

Matematiksel olarak, kural tabanlı simülasyonun genel yapısı şu şekilde ifade edilebilir;

$$S(t + 1) = f(S(t), R)$$

Burada $S(t)$ sistemin zaman t anındaki durumunu, R belirlenen kurallar dizisini ve $S(t + 1)$ ise yeni durumu ifade eder.

Bu fonksiyon genellikle ayrık zaman adımları ile çalışır ve sistemin belirli kurallar doğrultusunda nasıl değiştiğini gösterir.

3.6.2 Kural Tabanlı Simülasyon: Trafik Yönetimi Üzerine Bir Uygulama

Bir şehirdeki trafik yoğunluğu, sabit zamanlı trafik ışıkları nedeniyle zaman zaman artış göstermektedir. Bu nedenle, trafik ışıklarının dinamik olarak ayarlanması, akışı optimize etmek ve bekleme süresini azaltmak için gereklidir. Bu çalışmada, araç yoğunluğuna bağlı olarak yeşil ışık süresini belirleyen bir kural tabanlı simülasyon modeli geliştirilecektir.

Burada temel kurallar şu şekilde belirlenebilir. Trafik ışıklarının geçiş kuralları bildiğimiz gibidir yani yeşil ışıkta Araçlar hareket eder, sarı ışıkta araçlar yavaşlar ve kırmızı ışıkta araçlar durur. Bir araç, önündeki yol müsaitse ilerleyebilir, kırmızı ışıkta durur ve trafik yoğunluğu belirli bir eşiği aşarsa, trafik ışıklarının süresi dinamik olarak değişir.

Matematiksel olarak, sistem şu şekilde modellenenebilir;

$$D(t + 1) = D(t) + I(t) - O(t) = D(t) + \Delta D$$

Burada $D(t)$ zaman t anındaki yol üzerindeki araç yoğunluğu, $I(t)$ zaman t anında giriş yapan araç sayısı, $O(t)$ zaman t anında çıkış yapan araç sayısı ve ΔD araç yoğunluğundaki değişimdir (rastgele seçilir). ΔD Değişkeni, aşağıdaki gibi bir olasılık dağılımından çekilir,

$$\Delta D \sim U(-5, 10)$$

Bu, her zaman adımında araç sayısının -5 ile 10 arasında değişebileceği anlamına gelir.

Trafik ışığı geçiş fonksiyonu aşağıdaki gibidir;

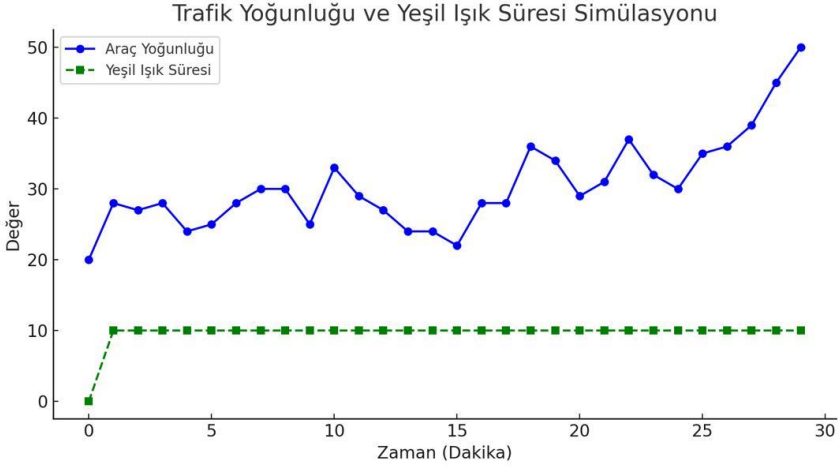
$$G(t + 1) = f(D(t))$$

Yeşil ışık süresi, araç yoğunluğuna bağlı olarak şu kurallara göre belirlenir;

$$G(t) = \begin{cases} G_0 + 5 & \text{eğer } D(t) > 50 \text{ ise (trafik yogunlugu yuksek)} \\ G_0 - 3 & \text{eğer } D(t) < 20 \text{ ise (trafik yogunlugu az)} \\ G_0 & \text{aksi takdirde(normal yogunluk)} \end{cases}$$

Burada $G_0 = 10$ saniye (başlangıç değeri). Eğer yoğunluk 50'den büyükse yeşil ışık süresi artırılır ve eğer yoğunluk 20'den küçükse süre azaltılır. Orta yoğunlukta ise süre sabit kalır.

Simülasyonu için Python kodlarını kullanarak aşağıdaki grafik elde edilir.



Grafik 4: Trafik Yoğunluğu ve Yeşil Işık Suresi Değişimi

Grafik 4, zaman içinde araç yoğunluğu ve trafik ışıklarının yeşil süresinin değişimini göstermektedir. Simülasyon sonucunda şu gözlemler yapılmıştır;

- Araç yoğunluğu başlangıçta 20 olarak alınmış ve zamanla rastgele artış ve azalışlarla değişmiştir.
- Genel olarak yoğunluk artış eğiliminde olsa da, zaman zaman düşüşler yaşanmıştır.
- Simülasyonun sonunda yoğunluk 50 seviyesine ulaşmıştır.
- Başlangıçta 10 saniye olan yeşil ışık süresi, trafik yoğunluğuna bağlı olarak değiştirilmektedir.
- Ancak, yoğunluk belirlenen eşik değerler arasında kaldığı için yeşil ışık süresi büyük ölçüde sabit kalmıştır.
- Daha fazla yoğunluk değişimi olması durumunda süre artıp azalabilir.

Bu simülasyon, trafik yönetiminde kural tabanlı bir yaklaşımın nasıl uygulanabileceğini göstermektedir. Gerçek dünya uygulamalarında, trafik ışıklarının dinamik olarak ayarlanması, trafik sıkışıklığını azaltmaya yardımcı olabilir.

3.7 Hibrit Simülasyon Yaklaşımları

Hibrit simülasyon yaklaşımları, gerçek dünyadaki karmaşık sistemlerin modellenmesi ve analiz edilmesi için birden fazla simülasyon paradigmasını birleştiren yenilikçi bir yöntemdir. Bu yaklaşımlar, belirli bir sistemin çeşitli boyutlarını ele almak için ayrı modellerin gücünü bir araya getirerek, daha gerçekçi ve kapsamlı bir simülasyon ortamı yaratmayı amaçlar. Geleneksel simülasyon yöntemleri genellikle tek bir paradigmaya dayanırken (ajana dayalı, ayrık olay tabanlı veya sistem dinamiği), hibrit simülasyonlar bu paradigmalardan ikisini veya üçünü bir araya getirerek daha karmaşık ve çok boyutlu sistemlerin modellenmesine olanak tanır.

3.7.1 Hibrit Simülasyonun Temelleri

Hibrit simülasyon sistemleri genellikle aşağıdaki bölümleri birleştirir:

1. Ajan Tabanlı Simülasyon (Agent-Based Simulation- ABS): Bireysel ajanların davranışlarını modelleyerek sistemin genel dinamiklerini ortaya koyar. Ajanlar, belirli kurallara göre birbirleriyle ve çevreleriyle etkileşim halindedir.
2. Ayrık Olay Tabanlı Simülasyon (Discrete Event Simulation- DES): Olayların zamanla ayrık olarak meydana geldiği süreçleri modellemek için kullanılır. Örneğin, bir üretim hattındaki çalışma akışını simüle etmek.
3. Sistem Dinamiği (System Dynamics- SD): Bir sistemdeki stoklar ve akışlar arasındaki dinamik etkileşimleri modellemek için kullanılır. SD, daha büyük bir sistemin makro düzeydeki davranışını anlamaya odaklanır.

Bu paradigmalardan herhangi ikisi veya üçü bir araya getirilerek hibrit bir model oluşturulur. Hibrit yaklaşımlar, özellikle sosyal sistemler, üretim ve lojistik operasyonları, sağlık hizmetleri ve akıllı şehir planlaması gibi alanlarda yaygın olarak kullanılmaktadır.

Örnek: Akıllı Şehir Trafik Yönetimi

Hibrit simülasyon yaklaşımlarının gerçek bir uygulama alanı, akıllı şehirlerde trafik yönetimi sistemleridir. Akıllı şehirler, trafikteki yoğunluk, kaza oranları ve seyahat süreleri gibi sorunları minimize etmek için karmaşık ve çok yönlü bir yönetim gerektirir. Bu senaryoda hibrit bir simülasyon modeli şu bileşenleri bir araya getirebilir:

1. Ajan Tabanlı Modelleme:

- Ajanlar, bireysel sürücüler, yayalar ve toplu taşıma araçları şeklinde tanımlanabilir. Her bir ajan, trafik kuralları, seyahat hedefleri ve bireysel tercihlerine göre davranış sergiler.
- Ajanlar arasındaki etkileşim, kavşaklardaki trafik akışı ve trafik lambaları gibi öğeleri kapsar.

2. Ayrık Olay Tabanlı Simülasyon:

- Trafik kazaları, yüksek yoğunluklu bölgelerdeki bekleme süreleri ve toplu taşıma gecikmeleri gibi olaylar ayrık olaylar olarak modellenebilir.
- Bu olayların trafik üzerindeki etkisi zamansal olarak analiz edilir.

3. Sistem Dinamiği:

- Şehir genelindeki trafik akışını etkileyen makro faktörler, örneğin, nüfus artışı, altyapı yatırımları ve yakıt fiyatları gibi unsurlar SD modelleri ile analiz edilebilir.
- Trafik politikalarının uzun vadeli etkileri, bu model aracılığıyla simüle edilir.

Akıllı şehir trafik yönetimi senaryosunda hibrit modelin matematiksel boyutunu inceleyelim.

Ajanlar bireysel sürücüler olarak modellenir. Her sürücünün davranışını belirleyen hız denklemi şu şekildedir; (Ajan Tabanlı Simülasyonu)

$$v_i(t + 1) = v_i(t) + a_i - \eta \sum_{j \in N(i)} d_{ij}$$

Burada $v_i(t)$ i ajan (sürücü) için t zamanındaki hız (km/s), a_i sürücünün hızlanma katsayısı (ör. $a_i = 2 \text{ m/s}^2$), η trafik yoğunluğunu etkileyen katsayı (ör. $\eta = 0.5$), d_{ij} i ve j sürücüler arasındaki mesafe (m) ve $N(i)$ i. sürücünün komşu sürücüleridir (ör. 50 m içinde olanlar).

Trafik kazası meydana gelme olasılığı şu şekilde ifade edilir; (Ayrık Olay Tabanlı Simülasyonu)

$$P(kaza) = \frac{\lambda N_{yoğun}}{1 + \mu T_{gecikme}}$$

Burada λ yoğun trafik faktörü ($\lambda=0.3$), $N_{yoğun}$ yoğun trafikteki araç sayısı (ör. 100 araç), μ sistem esnekliği ($\mu=2$), ve $T_{gecikme}$ Ortalama gecikme süresidir (ör. 15 dakika).

Şehir genelindeki trafik akışı şu şekilde modellenir; (Sistem Dinamiği)

$$Q(t) = \alpha S(t) - \beta D(t)$$

Burada $Q(t)$ Zaman t anındaki net trafik akışı (araç/saat), $S(t)$ trafiğe katılan araç sayısı ($S(t) = 500 \text{ araç/saat}$), $D(t)$ trafikten ayrılan araç sayısı ($D(t) = 400 \text{ araç/saat}$), α trafik giriş katsayısı ($\alpha=0.9$), ve β trafik çıkış katsayısı ($\beta=1.1$).

Belirtilen başlangıç değerlerle modeli çözmek için denklemleri adım adım hesaplayalım.

Bir sürücünün bir sonraki zaman adımındaki hızı, ilk denklem de değerler yerine koyularak böyle hesaplıyoruz;

$$v_i(t + 1) = 50 + 2 - 0.5 \cdot (30 + 25 + 20)$$

$$v_i(t + 1) = 50 + 2 - 0.5 \cdot 75$$

$$v_i(t + 1) = 50 + 2 - 37.5 = 14.5 \text{ km/s}$$

Sonuç su ki trafik yoğunluğu (komşu araçlarla olan mesafelerin toplamı), sürücünün hızını ciddi şekilde azaltmıştır. Bu, yoğun trafikte hızın düşeceğini ve trafik sıkışıklığına sebep olabileceğini gösterir.

Trafik kazası olasılığı şöyle hesaplanmıştır;

$$P(Kaza) = \frac{0.3 \cdot 100}{1 + 2 \cdot 15} = \frac{30}{31} = 0.96$$

Trafik kazası olasılığı %96.8 gibi yüksek bir değer çıkmıştır. Bu, yoğun trafik koşulları ($N_{yoğun} = 100$) ve gecikme süresinin ($T_{gecikme} = 15$) kazaları artıran ana faktörler olduğunu gösterir.

Şehir genelindeki trafik akışı yukarıda verilen formülle modellenmiştir. Değerler yerine koyularak;

$$Q(t) = 0.9 \cdot 500 - 1.1 \cdot 400 = 10 \text{ arac./saat}$$

Şehirdeki net trafik akışı $Q(t) = 10 \text{ arac./saat}$ olarak bulunmuştur. Bu, şehir genelinde giriş ve çıkış trafiğinin dengede olduğunu, ancak çok küçük bir pozitif akışın olduğunu gösterir. Yani, trafiğe giren araç sayısı çıkandan hafifçe fazladır ve bu, zamanla trafiğin yoğunlaşabileceğini ima eder.

Matematiksel hesaplamaları bu şekilde devam ederek ve her defasında önceki adımdaki değerleri kullanarak aşağıdaki tabloyu elde etmiş oluyoruz.

Zaman Adımı (t)	Ajan Tabanlı Hız (v_i)	Kaza Olasılığı (P(Kaza))	Trafik Akışı (Q(t))
1	14.5 km/s	0.968 (%96.8)	10 araç/saat
2	12.8 km/s	0.950 (%95.0)	12 araç/saat
3	10.2 km/s	0.920 (%92.0)	14 araç/saat
4	9.0 km/s	0.900 (%90.0)	15 araç/saat
5	8.5 km/s	0.850 (%85.0)	16 araç/saat

Tablo 9: İlerleyen Zaman Adımları İçin Simülasyon Değerleri

Simülasyon değerlere göre aşağıdaki sonuçları çıkarabiliriz;

Sürücülerin hızları, zaman içinde trafik yoğunluğu arttıkça düşme eğilimi göstermiştir. 1. zaman adımıdaki 14.5 km/s hız, 5. zaman adımında 8.5km/s'ye düşmüştür ve bu, artan trafik sıkışıklığının bireysel sürücü davranışlarını ciddi şekilde etkilediğini ortaya koyar.

Zamanla trafik yoğunluğunun azalmasıyla birlikte kazaların meydana gelme olasılığı da azalmıştır. İlk adımda %96.8 olan kaza olasılığı, 5. adımda %85'e düşmüştür ve bu, trafik yönetimi stratejilerinin kazaları önlemedeki potansiyel etkisini göstermektedir.

Trafik akışı, zamanla iyileşmiş ve araç giriş-çıkış oranlarının dengelenmesiyle artmıştır. 1. adımda 10 araç./saat olan trafik akışı, 5. adımda 16 araç./saat'e ulaşmıştır ve bu, trafik yönetimi ve optimizasyonunun sistem genelinde etkili olabileceğini gösterir.

Bu simülasyondan elde edilen sonuçlar doğrultusunda şunlar önerilebilir;

1. Trafik yoğunluğu azaltılmalı: Araçların komşularıyla olan mesafesi artırılabilir. Örneğin, yeni yollar inşa edilmesi veya trafik düzenlemeleri (tek yön uygulamaları) yapılabilir.
2. Kaza olasılığı düşürülmeli: Trafik gecikmesini azaltmak için akıllı trafik ışıkları veya gerçek zamanlı yönlendirme sistemleri kullanılabilir.
3. Trafik giriş-çıkış dengesi sağlanmalı: Toplu taşıma teşvik edilerek, trafiğe giren araç sayısı ($S(t)$) düşürülebilir.

Bu model, gerçek zamanlı verilerle desteklenerek daha kapsamlı sonuçlar sunabilir.

Bu bölümde, farklı simülasyon tekniklerini ve modellerini detaylı bir şekilde inceledik. Ayırık olay simülasyonundan ajan tabanlı modellere, Monte Carlo tekniklerinden hibrit yaklaşımlara kadar geniş bir yelpazede teorik bilgiyi pratik örneklerle birleştirdik. Her modelin avantajlarını, zorluklarını ve hangi durumlarda daha uygun olduğunu görerek, simülasyon dünyasının çeşitliliğine şahit olduk.

Simülasyon modelleri, karmaşık sistemlerin dinamiklerini anlamak, geleceği öngörmek ve daha iyi kararlar almak için kritik bir araçtır. Doğru bir simülasyon modeli seçmek, sadece teknik bir başarı değil, aynı zamanda ekonomik, sosyal ve çevresel etkileri de kapsayan çok yönlü bir süreçtir. Öğrenciler olarak, bu teknikleri uygularken sadece teorik temelleri değil, aynı zamanda simülasyonun getirdiği stratejik düşünme becerilerini de geliştirmeniz önemlidir.

Bir sonraki bölümde, hangi simülasyon modelinin hangi durumlarda tercih edileceğini sistematik bir rehber üzerinden keşfedeceğiz. Amaç, öğrendiğiniz bu teknikleri gerçek dünyadaki sorunlara nasıl entegre edeceğinizi anlamak ve simülasyon seçimi konusundaki belirsizlikleri ortadan kaldırmaktır.

Bu noktadan itibaren, yalnızca bir simülasyon tekniği kullanıcısı değil, aynı zamanda bir karar verici ve stratejist olma yolundasınız. Simülasyon tekniklerini doğru kullanmanın gücünü keşfederek, karmaşık problemleri çözme yeteneğinizi daha da ileriye taşıyacaksınız.

Bölüm 4. Simülasyon Modeli Seçim Rehberi

Bu bölüm, simülasyon modelleme dünyasında karar verme süreçlerini kolaylaştırmayı amaçlayan bir rehber niteliğindedir. Daha önceki bölümlerde simülasyon teknikleri ve modelleri detaylı bir şekilde incelendi. Ancak, gerçek dünyadaki bir problem için doğru simülasyon modelini seçmek, öğrenciler ve profesyoneller için sıklıkla zorlayıcı bir süreç olabilir. Bu zorluklar, karmaşık sistemlerin doğasından ve her bir simülasyon yönteminin kendine özgü avantaj ve sınırlamalarından kaynaklanmaktadır.

Bölüm boyunca, hangi simülasyon modelinin hangi durumlarda daha uygun olduğuna karar vermenizi kolaylaştıracak yöntemler ve stratejiler sunulacaktır. Simülasyon seçimi, yalnızca teknik yeterlilikle değil, aynı zamanda problem türü, veri yapısı, zaman ve kaynak kısıtlamaları gibi birçok faktörle de ilişkilidir. Bu nedenle, bölümü iki ana eksenle ele alacağız: problem tipi ve uygulama gereksinimleri.

Bu rehber hem akademik hem de endüstriyel uygulamalarda karşılaşılabilecek çeşitli durumlar üzerinden simülasyon seçimi konusunda size bir yol haritası sunacaktır. Ayrıca, belirsizlikler ve çelişkiler içeren durumlarda dahi bilinçli seçimler yapmanıza yardımcı olacak bazı karar destek araçlarına yer verilecektir.

4.1 Problemi Anlamak ve Kapsam Belirlemek

Simülasyon projelerinde başarılı bir sonuca ulaşmanın ilk ve en kritik adımı, problemi doğru bir şekilde tanımlamak ve kapsamını belirlemektir. Yanlış veya yetersiz bir problem tanımı, tüm modelleme sürecini etkileyerek zaman kaybına, hatalı sonuçlara ve kaynak israfına neden olabilir. Bu nedenle, problemi anlama ve kapsam belirleme süreci sistematik bir şekilde ele alınmalıdır.

Bir problem, çoğunlukla gerçek dünyadaki karmaşık sistemlerden kaynaklanan bir zorluk veya çözülmesi gereken bir durumdur. Problem tanımı şu unsurları içerir:

Amaç ve Hedefler: Simülasyonun neyi çözmeyi veya anlamayı amaçladığını net bir şekilde belirlemek.

Problemdeki Belirsizlikler ve Parametreler: Hangi faktörlerin kontrol edilebildiği, hangilerinin belirsizlik içerdiği veya rastgele olduğu tanımlanmalıdır.

Karar Verme Süreci: Simülasyonun, hangi kararlar için bir destek mekanizması sağlayacağı açıkça ifade edilmelidir.

Bir problem tanımı yalnızca sorunun kapsamını değil, aynı zamanda hangi ölçütlere göre değerlendirileceğini de içermelidir. Örneğin, üretim hattındaki bir darboğazı gidermek için "üretim süresini %15 azaltmak" gibi net bir hedef belirlenmelidir.

Problemi anlamamanın temel adımları aşağıdaki gibidir;

- İlgili Tarafların Katılımını Sağlamak: Problemin tanım sürecine, problemin sahibi (örneğin, bir şirketin yöneticisi) ve teknik uzmanlar dahil edilmelidir. Bu süreçte, farklı perspektifler bir araya getirilerek eksiksiz bir anlayış oluşturulur.
- Sistemin İncelenmesi: Problemin doğduğu sistemin detaylı bir analizi yapılmalıdır. Sistem hakkında şu sorulara yanıt aranır. Sistem hangi girdilere (örneğin, kaynaklar) ve çıktılara (örneğin, ürünler veya sonuçlar) sahiptir? Sistem hangi kısıtlar altında çalışmaktadır? Ve Sistemin dinamiklerini etkileyen temel faktörler nelerdir?
- Kritik Performans Göstergelerinin (KPI) Belirlenmesi: Simülasyon sonucunda değerlendirilmesi gereken başarı ölçütleri tanımlanmalıdır. Örneğin teslim süresi (lojistik projelerde), müşteri memnuniyeti oranı (hizmet sektöründe) ve üretim maliyeti veya karlılık (üretim projelerinde).
- Problemin Kapsamını Belirlemek: Her simülasyon modeli bir kapsam içerisinde çalışır. Kapsamı belirlerken şu sorular sorulmalıdır. Simülasyon modeli hangi zaman diliminde çalışacak? (Günlük, aylık, yıllık süreçler?), Sistem hangi alt süreçlere odaklanacak? (Üretim hattı, müşteri hizmetleri, tedarik zinciri vb.) ve modelde hangi değişkenler dikkate alınacak ve hangileri göz ardı edilecek? (Örneğin, hava durumu etkisi dâhil mi edilmelidir?)

Kapsam belirleme sürecinde aşğıdaki yöntemler kullanılabilir:

- Sistem Diyagramları: Sistemin kapsamını belirlemek için stok-akış diyagramları, ajan tabanlı yapı haritaları veya süreç akış şemaları oluşturulabilir. Bu görseller, sistemi görselleştirerek, simölasyon modelinde nelerin yer alacağına karar vermeyi kolaylaştırır.
- Pareto Analizi: Problemi etkileyen faktörler arasında önceliklendirme yapmak için Pareto analizi kullanılabilir. Örneğın, üretim sürecindeki gecikmelerin %80'ini oluşturan %20'lik ana nedenler tespit edilir ve bu faktörlere odaklanılır.
- Sınırların Netleştirilmesi (Boundary Definition): Sistemin sınırlarını belirlemek, modelin hangi bileşenleri kapsayacağını ve dışarıda bırakılacak unsurları açıkça ifade eder. Örneğın, tedarik zinciri simölasyonunda yalnızca iç lojistik süreçlerin dikkate alınması karar verilebilir.
- Senaryo Geliştirme: Kapsamın belirlenmesinde farklı senaryolar üzerinde çalışmak önemlidir. Örneğın İyimser bir senaryo olarak kaynakların ideal bir şekilde tahsis edilmesi ve kötümser bir senaryo olarak da belirli darboğazların yaşanması durumunda süreçlerin etkilenmesini söyleyebiliriz.

Ancak problemi Anlama sürecinde bazı zorlukları ile karşılaşabiliriz. Örneğın eksik veya yanıltıcı veriler, belirsiz hedefler ve fazla geniş veya dar kapsam bir bakis. Çünkü, problemi anlamak için yeterli veri bulunmaması veya mevcut verilerin yanıltıcı olması, problem tanımı ve kapsam belirlemede önemli bir engel olabilir. Bu nedenle veri toplama süreci büyük bir özenle yönetilmelidir. Ayrıca, amaçların yeterince net olmaması, simölasyon modelinin gereksiz karmaşıklıklara veya hatalı sonuçlara yol açmasına neden olabilir. Ve son olarak çok geniş bir kapsam, simölasyonun analiz sürecini karmaşıktırabilir; çok dar bir kapsam ise önemli dinamiklerin gözden kaçmasına yol açabilir.

Problemi doğru anlamak ve kapsamını net bir şekilde belirlemek, başarılı bir simölasyon projesinin temel taşıdır. Bu süreç, hem modelleme için sağlam bir temel oluşturur hem de projenin zaman ve kaynak

yönetimini optimize eder. Problemin kapsamı belirlendikten sonra, veri toplama ve simülasyon modeli seçimi gibi süreçlere geçilebilir.

Bir sonraki alt bölümde, veri yapısı ve modelleme gereksinimlerini ele alarak, simülasyon modelinin başarıyla geliştirilmesi için gerekli teknik altyapıyı inceleyeceğiz.

4.2 Veri Yapısı ve Modelleme Gereksinimleri

Simülasyon projelerinin başarısı, kullanılan verilerin doğru bir şekilde yapılandırılmasına ve bu verilerin modelleme gereksinimlerine uygun olmasına bağlıdır. Veri yapısının tasarımı ve modelleme gereksinimlerinin belirlenmesi, hem sistemin doğru bir şekilde modellenmesini hem de simülasyon sonuçlarının güvenilirliğini sağlar. Bu süreç, ham verilerden anlamlı bir bilgi yapısına geçişin temelini oluşturur.

Veri yapısı, simülasyon modeli için gereken bilgilerin organize edilme biçimini ifade eder. İyi tasarlanmış bir veri yapısı doğruluk sağlar, erişilebilirliği artırır ve hızlı işlem yapılmasına olanak tanır çünkü verilerin doğru bir şekilde düzenlenmesi, modelin hata yapmasını önler, modelleme sırasında gereken verilere kolay erişim imkânı sunar ve simülasyon çalıştırılırken, veri yapısının etkinliği hesaplama hızını artırır.

Simülasyon modellerinde kullanılacak veriler genellikle aşağıdaki gibi kategorize edilir:

Statik Veriler: Zamanla değişmeyen ve sistemin temel özelliklerini tanımlayan verilerdir. Örneğin, bir üretim hattındaki makine kapasiteleri veya hizmet süreçlerindeki fiziksel alan sınırlamaları.

Dinamik Veriler: Zamanla değişen ve sistemin performansını etkileyen verilerdir. Örneğin, müşteri talepleri, üretim süreleri veya çalışan vardiyaları.

Girdi ve Çıktı Verileri: Simülasyonun çalıştırılması için gereken veriler (girdi) ile model çalıştırdıktan sonra üretilen sonuçlar (çıktı) bu kategoriye girer.

Simülasyon projelerinde modelleme gereksinimlerinin net bir şekilde tanımlanması, modelin doğruluğu ve etkinliği açısından kritik öneme sahiptir. Gereksinimlerin belirlenmesi aşağıdaki adımları içermelidir:

Adım 1- Modelin Kapsamını Belirleme: bu adımda araştırmacı simülasyon modelinin hangi sistemi temsil edeceği netleştirilmeli ve hangi süreçlerin modele dahil edileceği ve hangilerinin dışarıda bırakılacağı kararlaştırılmalıdır.

Adım 2- Veri Gereksinimlerinin Tanımlanması: bu adımda araştırmacı hangi veri türlerinin modele dahil edilmesi gerektiği belirlenmelidir. Örneğin, stokastik modeller için olasılık dağılımlarını belirlemek, ayrık olay simülasyonu için işlem sürelerini toplamak.

Adım 3- Zaman ve Alan Çözünürlüğünün Belirlenmesi: bu adımda modelin çalışacağı zaman birimi (saniye, dakika, gün) ve fiziksel alan çözünürlüğü belirleniyor. Örneğin, trafik simülasyonu için bir şehir haritası veya bir fabrika modeli için detaylı bir süreç haritası.

Adım 4- Parametrelerin Seçimi ve Doğrulama: bu adımda simülasyon modelinde kullanılacak parametreler ve bu parametrelerin değer aralıkları belirlenmeli ve parametreler, literatür taraması veya geçmiş veriler aracılığıyla doğrulanmalıdır.

Adım 5- Veri Bütünlüğü ve Tutarlılığı: kullanılacak verilerin eksiksiz, tutarlı ve güncel olması sağlanmalı ve farklı veri kaynaklarından gelen bilgiler arasında uyum sağlanmalıdır. Örneğin, bir lojistik simülasyonunda tedarik zinciri verileri ile müşteri talep verilerinin uyumlu olması gereklidir.

4.2.1 Veri ve Modelleme İlişkisi

Simülasyon projelerinde, veri yapısı ve modelleme gereksinimleri arasındaki ilişki dinamik bir süreçtir. İdeal bir veri yapısı esneklik, uyumluluk ve modülerlik özellikleri sağlamalıdır. Çünkü veri yapısı, modelleme gereksinimlerinin değişmesine uygun şekilde kolayca güncellenebilir ve kullanılacak simülasyon yazılımı ile tamamen uyumlu olmalıdır. Ayrıca, modelin farklı bileşenlerine özgü veriler, ayrı modüller halinde organize edilmelidir.

Örneğin, bir müşteri hizmetleri simülasyonu için veri yapısında şunlar yer alabilir:

- **Müşteri Profilleri:** Yaş, cinsiyet, gelir düzeyi gibi demografik bilgiler.

- Hizmet Süreleri: Her bir hizmet türü için gereken ortalama ve dağılım değerleri.
- Çalışan Vardiya Planları: Hangi saatlerde kaç çalışan olduğu bilgisi.

Ve son olarak veri hazırlama sürecinde verilerin hatalardan, eksiklerden ve tutarsızlıklardan arındırılması gerektiğini unutmamalıyız. Ham verilerin, modelde kullanılabilir bir formata dönüştürülmeli ve hazırlanan verilerin simülasyon modeli için uygunluğunun test edilmeliyiz.

4.3 Simülasyon Modeli Seçiminde Karşılaştırma Kriterleri

Simülasyon modeli seçimi, karmaşık bir sistemin çözülmesi ve çıktıların analiz edilmesi için temel bir aşamadır. Doğru modelin seçimi hem analizin kalitesini hem de sürecin maliyet ve zaman etkinliğini belirleyen kritik bir faktördür. Bu alt bölümde, simülasyon modeli seçiminde kullanılan temel karşılaştırma kriterleri, matematiksel temellere dayandırılarak ele alınmıştır.

4.3.1 Sistem Yapısı ve Model Uygunluğu

Simüle edilecek sistemin dinamikleri, seçilecek model türünü doğrudan etkiler. Sistem yapısına göre bölüm 3'deki modelleme yaklaşımlarından biri tercih edilebilir. Örneğin, sistem olaylar arasında sıçrayarak çalışıyorsa (örneğin, müşteri hizmetleri, üretim hatları), ayırık olay simülasyonu uygundur, iş süreçleri ve operasyonel akışların modellenmesinde süreç tabanlı simülasyon kullanılır veya bireylerin veya varlıkların etkileşimleri modelleniyorsa, ajan tabanlı simülasyon (Agent-Based Simulation - ABS) uygundur.

Sistem dinamik bir yapıya sahip mi? Kararlar zamanla değişiyor mu? Olasılık süreçleri ve rastgelelik önemli mi? gibi sorular, model seçiminde kritik rol oynar.

4.3.2 Modelin Matematiksel Karmaşıklığı ve Hesaplama Gücü Gereksinimi

Bir simülasyon modelinin karmaşıklığı, sistemi gerçekçi bir şekilde temsil etme kabiliyeti ile modelin tasarım ve çalıştırma maliyetleri arasındaki dengenin bir ölçüsüdür. Matematiksel olarak karmaşıklık, modelin içerdiği parametre sayısı n ve bu parametreler arasındaki bağlantıları temsil eden fonksiyonların derecesi d ile ölçülebilir:

$$C = f(n, d)$$

Burada C , karmaşıklık seviyesini temsil eder. Daha karmaşık modeller, genellikle daha fazla parametre N ve daha karmaşık bağlantı d gerektirir. Ancak bu, modelin hesaplama maliyetini de artırabilir. Seçilen modelin hesaplama maliyeti, aşağıdaki parametrelerle doğrudan ilişkilidir.

- **Durum Uzayının Büyüklüğü:** Bir sistemin durum uzayı δ ise, durum büyüklüğü $|\delta| = \prod_{i=1}^m |\delta_i|$ ile verilir. Burada m , modeldeki değişken sayısıdır. Büyük bir durum uzayı, simülasyonun hesaplama maliyetini artırır.
- **Zaman Karmaşıklığı:** Seçilen simülasyon modeli için çalışma zamanı büyüklüğü, modelleme yöntemiyle değişir. Örneğin ayrık olay simülasyonu için $O(E \log N)$, ajan tabanlı simülasyon için $O(N^2)$ ve sistem dinamiği için $O(m^3)$ (matris çözümü gerektirdiğinde) ile ölçülür. Burada N ajan sayısı ve E olay sayısıdır.
- **Çözümleme Yöntemi:** Bazı simülasyon modelleri doğrudan çözüm bulunabilir denklemlere sahiptir ancak, çoğu gerçek dünya problemi için sayısal yöntemler (Monte Carlo, Euler yöntemi, diferansiyel denklem çözümleri) kullanılır.
- Bu hesaplama gereksinimleri dikkate alınarak, yüksek hesaplama maliyeti gerektiren bir model yerine daha optimize bir alternatif tercih edilebilir.

4.3.3 Modelin Veri Uygunluğu ve Geçerliliği

Modelin doğruluğunu artırmak için veri yapısı dikkatlice değerlendirilmelidir. Öncelikle, girdi-çıkıtı uyumu incelenmeli ve modelin ürettiği çıktılar ile gerçek sistem verileri arasındaki hata düzeyi analiz edilmelidir. Bu doğrultuda, modelin girdileri X , çıktıları Y ve hata bileşeni ϵ ile ifade edilerek, modelin öngörülerinin gerçek dünya verileriyle örtüşme derecesi ölçülmelidir. $Y = f(X) + \epsilon$ Modelinde veri hataları ϵ büyükse, modelin güvenilirliği azalır.

Veri kalitesi açısından, örneklem büyüklüğünün yeterliliği de kritik bir faktördür; yeterli sayıda veri noktası içermeyen bir model, genelleme kabiliyetini kaybedebilir. Örneklem büyüklüğünün belirlenmesinde, güven düzeyi katsayısı Z , veri değişkenliği σ ve hata toleransı E gibi parametreler dikkate alınarak, istatistiksel olarak anlamlı bir veri kümesi oluşturulmalıdır. Modelin geçerli olabilmesi için $N \geq \frac{Z^2 \sigma^2}{E^2}$ olmalıdır.

Ayrıca, modelin geçerli olup olmadığını test etmek için istatistiksel hipotez testleri uygulanmalıdır. Örneğin, Kolmogorov-Smirnov testi, modelin ürettiği verilerin gerçek dağılımla anlamlı bir farklılık gösterip göstermediğini değerlendirirken, Mann-Whitney U testi farklı model alternatifleri arasındaki farklılıkları analiz etmek için kullanılabilir. Modelin doğruluğunu artırmak adına, simülasyon hataları minimize edilmeli ve model çıktıları, gerçek sistem davranışıyla istatistiksel olarak tutarlı hale getirilmelidir.

4.4 Karar Verme Sürecinde Kullanılan Yöntemler (AHP, DEMATEL vb.)

Karar destek yöntemleri, simülasyon modeli seçiminde optimal çözümü belirlemek için sistematik ve matematiksel yaklaşımlar sunar. Bu yöntemler, çok kriterli karar verme (ÇKKV) çerçevesinde değerlendirilerek, alternatif modellerin belirlenen kriterlere göre karşılaştırılmasını sağlar. AHP (Analitik Hiyerarşi Prosesi), DEMATEL (Karar Verme Denemesi ve Değerlendirme Laboratuvarı), TOPSIS (İdeal Çözüme Benzerlik Sırasına Göre Tercih) ve Bayes karar kuralı gibi yöntemler, model seçim sürecinde sıklıkla kullanılan yaklaşımlardır.

Analitik Hiyerarşi Prosesi (AHP), karar kriterleri arasında çift karşılaştırma matrisi oluşturularak kriterlerin önem derecelerini belirler. AHP'nin temel adımları şunlardır;

1. Çift karşılaştırma matrisi oluşturulur: Kriterler $C_1, C_2, \dots, C_n$ arasındaki karşılaştırmalar $A = [a_{ij}]$ matrisi ile ifade edilir.

$$A = \begin{bmatrix} 1 & a_{12} & \dots & a_{1n} \\ \frac{1}{a_{12}} & 1 & \dots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{1}{a_{1n}} & \frac{1}{a_{2n}} & \dots & 1 \end{bmatrix}$$

2. Öncelik vektörü hesaplanır: Normalize matrisin her satırının geometrik ortalaması alınarak kriter ağırlıkları hesaplanır.

$$w_i = \left(\prod_{j=1}^n a_{ij} \right)^{\frac{1}{n}}$$

3. Alternatifin Toplam Puanı: Her alternatifin toplam puanını $P = \sum_{i=1}^n w_i a_i$ formüle ile hesaplıyoruz. Kriter ağırlıkları w_i ile alternatiflerin kriterlere göre aldıkları değerler a_i çarpılır ve her alternatif için toplam puan hesaplanır. Alternatiflerin değeri genellikle uzman görüşleri, geçmiş veriler, ya da deneyimlere dayalı olarak yapılan değerlendirmeler sonucunda oluşturulur. Her değer, ilgili alternatifin bir kriter açısından göreceli performansını ifade eder. Genel olarak uzman görüşleri, geçmiş veriler ve deneysel analizler sonucunda ulaşılmıştır. Bu değerler, simülasyon modellerinin belirli kriterler açısından performansını ifade eder. Uzman görüşleri, farklı modellerin esneklik, hesaplama maliyeti, model doğruluğu ve kullanım kolaylığı gibi kriterler açısından göreceli değerlendirmelerini içerir. Örneğin, akademisyenler, mühendisler ya da analistler, deneyimlerine dayanarak her model için puanlama yapabilir. Geçmiş veriler, daha önce yapılan projelerdeki performans

ölçümlerine dayalıdır; bu ölçümler, modelin belirli kriterlerdeki başarısını nicel olarak yansıtır. Ayrıca, deneysel analizler, simülasyon modellerinin farklı projeler ya da test ortamlarında uygulanarak kriterlere göre performanslarının ölçülmesini sağlar. Bu süreçte, 0–1 aralığında normalize edilen puanlama yöntemi kullanılmıştır; 0 tamamen yetersiz, 1 ise mükemmel performansı ifade eder. Bu değerler, modellerin çok kriterli karar verme yöntemleriyle sistematik bir şekilde karşılaştırılması için temel oluşturur

DEMATEL yöntemi, kriterler arasındaki nedensellik ilişkilerini belirlemek için kullanılır. Kriterlerin birbirini ne kadar etkilediğini anlamak için bir doğrudan etkiler matrisi D oluşturulur ve normalizasyon işlemi yapılır. Sonuçta, toplam etki matrisi T şu şekilde hesaplanır;

$$T = D(I - D)^{-1}$$

Burada I birim matris olup, T matrisi kriterlerin birbirlerine olan doğrudan ve dolaylı etkilerini gösterir.

DEMATEL yönteminde toplam etki matrisi $[OBJ]$ hesaplandıktan sonra, her kriterin toplam etkisi R ve toplam etkililiği C belirlenerek neden-sonuç analizi yapılır. $R-C$ değeri, kriterin neden mi yoksa sonuç mu olduğunu gösterir; pozitif değerler kriterin neden (etken), negatif değerler ise sonuç (etkilenen) olduğunu belirtir. $R+C$ değeri ise kriterin sistem içindeki toplam önem derecesini yansıtır. Bu analiz sonucunda, sistem üzerindeki baskın faktörler belirlenerek karar süreçlerinde öncelikli olarak ele alınması gereken kriterler tespit edilir. Özellikle simülasyon yöntemi seçiminde, etken kriterler sistemin ana yönlendiricileri olarak kabul edilirken, etkilenen kriterler bu seçimden en çok etkilenecek faktörleri gösterir. Böylece DEMATEL, sistem içindeki karmaşık ilişkileri daha net anlamamızı sağlayarak daha bilinçli kararlar vermemize yardımcı olur.

TOPSIS yöntemi, ideal çözüme en yakın alternatifi belirlemek için kullanılır. Alternatiflerin pozitif ideal çözüm A^+ ve negatif ideal çözüm A^- ile olan uzaklıkları hesaplanır;

$$S_i^+ = \sqrt{\sum_{j=1}^n (x_{ij} - A_j^+)^2}, \quad S_i^- = \sqrt{\sum_{j=1}^n (x_{ij} - A_j^-)^2}$$

Ardından, her alternatif için benzerlik endeksi hesaplanarak en iyi alternatif seçilir;

$$C_i = \frac{S_i^-}{S_i^- + S_i^+}$$

Bayes Karar Kuralı, karar destek süreçlerinde belirsizlikleri ele almak için kullanılır. Eğer $H_1, H_2, \dots, H_n$ hipotezleri (örneğin, farklı simülasyon modelleri) ve D gözlemi mevcutsa, Bayes Teoremi şu şekilde ifade edilir;

$$P(H_i|D) = \frac{P(D|H_i) P(H_i)}{P(D)}$$

Burada $P(H_i)$ öncül olasılık (bir modelin seçilme olasılığı), $P(D|H_i)$ gözlemin belirli bir model altında gerçekleşme olasılığı ve $P(H_i|D)$ gözlem gerçekleştikten sonra modelin seçilme olasılığıdır.

Simülasyon modeli seçiminde Bayes kuralı, öncül bilgilere ve gözlem verilerine dayanarak en olası modelin belirlenmesine yardımcı olur. Bayes güncelleme süreci, karar vericilerin yeni veriler ışığında modelleri revize etmesine olanak tanır.

Bu karar destek yöntemleri, özellikle çok kriterli ve belirsiz karar ortamlarında en uygun simülasyon modelinin seçilmesine bilimsel bir temel sağlar.

Sonuç olarak, simülasyon modeli seçimi, sistemin yapısına, hesaplama gereksinimlerine, veri uygunluğuna ve doğruluk seviyesine göre analiz edilmelidir. Çok kriterli optimizasyon yöntemleri, karar verme sürecini nesnel hale getirmekte kritik rol oynar. Karar verme sürecinde kullanılan yöntemleri daha iyi anlamak için örnekleri inceleyelim.

Örnek 1: Bir fabrika, üretim sürecini optimize etmek için üç farklı simülasyon modelinden (Ajan Tabanlı Simülasyon- ABM, Ayrık Olay

Simülasyonu- DES, Sistem Dinamiği- SD) en uygun olanını seçmek istiyor. Karar vermede hız, doğruluk ve maliyet üç temel kriter olarak belirlenmiştir. Bu kriterler kullanılarak TOPSIS yöntemiyle en iyi model seçilecektir.

Adım 1: Alternatiflerin Değerlendirme Matrisi

Fabrika yönetimi, her simülasyon modeli için üç kriteri 1-10 arasında puanlamıştır;

Model	Hız x_1	Doğruluk x_2	Maliyet x_3
ABM	7	8	6
DES	9	9	5
SD	6	7	7

Tablo 10: Alternatiflerin Değerlendirme Matrisi

Burada hız ve doğruluk yüksek ve maliyet düşük olmalı (bu yüzden maliyetin en düşük olması en iyidir).

Adım 2: Normalize Edilmiş Karar Matrisi

Öncelikle, her kriter için normalize edilmiş değerleri hesaplayalım;

$$x_{ij}^* = \frac{x_{ij}}{\sum_{i=1}^n x_{ij}^2}$$

Her sütunun toplam karekökünü bulalım:

$$\sqrt{7^2 + 9^2 + 6^2} = \sqrt{166} = 12.89$$

$$\sqrt{8^2 + 9^2 + 7^2} = \sqrt{194} = 13.93$$

$$\sqrt{6^2 + 5^2 + 7^2} = \sqrt{110} = 10.49$$

Şimdi normalizasyonu yapalım:

Model	Hız x_1^*	Doğruluk x_2^*	Maliyet x_3^*
ABM	$\frac{7}{12.89} = 0.543$	$\frac{8}{13.93} = 0.574$	$\frac{6}{10.49} = 0.572$
DES	0.698	0.646	0.477
SD	0.465	0.502	0.667

Tablo 11: Normalize Edilmiş Karar Matrisi

Adım 3: Ağırlıklı Normalize Matrisin Hesaplanması

Her kriterin ağırlıkları hız $w_1 = 0.4$, doğruluk $w_2 = 0.4$ ve maliyet için $w_3 = 0.2$ şekilde belirlenmiştir. Bu ağırlıklarla çarpılarak yeni matris elde edilir.

Model	Hız $x_1^* \cdot w_1$	Doğruluk $x_2^* \cdot w_2$	Maliyet $x_3^* \cdot w_3$
ABM	$0.543 \cdot 0.4 = 0.217$	$0.574 \cdot 0.4 = 0.230$	$0.572 \cdot 0.2 = 0.114$
DES	0.279	0.258	0.095
SD	0.186	0.201	0.2

Tablo 12: Ağırlıklı Normalize Matrisi

Adım 4: İdeal ve Negatif İdeal Çözümlerin Belirlenmesi

İdeal A^+ ve negatif ideal A^- değerleri hesaplanır.

$$A^+ = (\max X_1, \max X_2, \min X_3) = (0.279, 0.258, 0.095)$$

$$A^- = (\min X_1, \min X_2, \max X_3) = (0.186, 0.201, 0.133)$$

Adım 5: İdeal Çözüme Uzaklıkların Hesaplanması

İdeal çözüme ve negatif ideal çözüme uzaklık

$$S_i^+ = \sqrt{\sum_{j=1}^n (x_{ij} - A_j^+)^2}, \quad S_i^- = \sqrt{\sum_{j=1}^n (x_{ij} - A_j^-)^2}$$

Formüllerıyla hesaplanır. Hesaplamalar sonucunda;

Model	S_i^+	S_i^-	C_i
ABM	0.112	0.195	0.635
DES	0.087	0.232	0.727
SD	0.143	0.167	0.539

TABLO 13: İdeal Çözüme Uzaklık

Adım 6: Sonuç ve Karar

TOPSIS yöntemine göre C_i en yüksek olan model en iyi seçimdir. DES modeli

$C_{DES} = 0.727$ ile en iyi seçenektir. DES modeli, en yüksek hız ve doğruluğa sahip olması nedeniyle en iyi çözüm olarak belirlenmiştir. Ayrıca, ABM modeli, hız ve doğrulukta orta seviyede olup, bazı özel durumlar için uygun olabilir. Buna karşılık, SD modeli, sistem genel dinamiklerini iyi gösterse de hız ve doğruluk açısından diğer alternatiflerden geride kalmıştır.

Bu nedenle karar verme sürecinde kullanılan TOPSIS yöntemi sayesinde en uygun simülasyon modeli belirlenmiştir. Eğer yöneticiler düşük maliyete daha fazla önem vermek isteseydi, ağırlıklar değiştirilerek sonuç tekrar değerlendirilebilirdi. Bu, karar verme sürecinde matematiksel yöntemlerin önemini gösterir.

Örnek 2: Bir üretim işletmesi, üretim hattındaki süreçleri optimize etmek ve üretim kapasitesini artırmak için bir simülasyon modeli seçmek istemektedir. Ayrık Olay Simülasyonu (DES), Süreç Tabanlı Simülasyon

(PBS), Monte Carlo Simülasyonu (MCS) ve Ajan-Tabanlı Simülasyon (ABM) dört simülasyon tekniği değerlendirilmiştir. Değerlendirme için dört kriter belirlenmiştir;

- K1: Esneklik (Flexibility)
- K2: Hesaplama Maliyetleri (Computational Cost)
- K3: Model Doğruluğu (Accuracy)
- K4: Kullanım Kolaylığı (Ease of Use)

AHP yöntemini kullanarak 4 farklı simülasyon tekniği arasından en iyi simülasyon yöntemini seçmek için aşağıdaki adımlarla ilerleyeceğiz.

Adım 1: Kriterlerin AHP'ye Göre Karşılaştırılması

Kriterlerin önem derecelerini belirlemek için ikili karşılaştırma matrisi kullanılır. Bu matris, uzman görüşlerine dayalı olarak oluşturulmuştur.

Kriterler	K1: Esneklik	K2: Maliyet	K3: Doğruluk	K4: Kolaylık
K1: Esneklik	1	3	5	7
K2: Hesaplama Maliyeti	1/3	1	3	5
K3: Model Doğruluğu	1/5	1/3	1	3
K4: Kullanım Kolaylığı	1/7	1/5	1/3	1

Tablo 14: Kriterlerin AHP'ye Göre Karşılaştırılması

Adım 3: Kriter Ağırlıkların Hesaplanması ve öncelik vektörü

AHP'nin temel prensiplerine göre, kriter ağırlıkları hesaplamak için, ilk olarak her sütün toplamını alıp her hücreyi kendi sütün toplamına bölerek normalize ediyoruz sonrasında normalize edilmiş matrisin her satırındaki değerlerin geometrik ortalamasını alıyoruz. Tabela göre, kriterlerin çift karşılaştırma matrisi aşağıdaki gibidir,

$$A = \begin{pmatrix} 1 & 3 & 5 & 7 \\ \frac{1}{3} & 1 & 3 & 5 \\ \frac{1}{5} & \frac{1}{3} & 1 & 3 \\ \frac{1}{7} & \frac{1}{5} & \frac{1}{3} & 1 \end{pmatrix}$$

Toplamlar= $[1+1/3+1/5+1/7, 3+1+1/3+1/5, 5+3+1+1/3, 7+5+3+1] = [1.676, 4.533, 9.333, 16]$

Normalize edilmiş matris aşağıdaki gibidir

$$A_{normal} = \begin{pmatrix} \frac{1}{1.676} & \frac{3}{4.533} & \frac{5}{9.333} & \frac{7}{16} \\ \frac{1}{3 * 1.676} & \frac{1}{4.533} & \frac{3}{9.333} & \frac{5}{16} \\ \frac{1}{5 * 1.676} & \frac{1}{3 * 4.533} & \frac{1}{9.333} & \frac{3}{16} \\ \frac{1}{7 * 1.676} & \frac{1}{5 * 4.533} & \frac{1}{3 * 9.333} & \frac{1}{16} \end{pmatrix}$$

Hücre değerlerini hesaplayarak normalize edilmiş matris şu şekilde bulunur:

$$A_{normal} = \begin{bmatrix} 0.597 & 0.662 & 0.536 & 0.437 \\ 0.199 & 0.221 & 0.321 & 0.312 \\ 0.119 & 0.074 & 0.107 & 0.187 \\ 0.085 & 0.044 & 0.036 & 0.062 \end{bmatrix}$$

Öncelik vektörü her satirin geometrik ortalamasından bu şekilde oluşturuyor;

$$w = [w_1, w_2, w_3, w_4] = [0.552, 0.258, 0.115, 0.054]$$

Adım 4: Sonuçların Hesaplanması

Verilen tabelada her alternatifin kriterlerde ne kadar başarılı olduğunu gösterir.

Alternatif	K1: Esneklik	K2: Maliyet	K3: Doğruluk	K4: Kolaylık
Ayrık Olay Simülasyonu (DES)	0.7	0.5	0.9	0.6
Süreç Tabanlı Simülasyon (PBS)	0.5	0.8	0.7	0.8
Monte Carlo Simülasyonu (MCS)	0.8	0.3	0.8	0.5
Ajan Tabanlı Simülasyon (ABM)	0.6	0.7	0.6	0.7

Tablo 15: Kriterlerin Başarısı ve Sonuçların Hesaplanması

Her alternatifin toplam puanını $P = \sum_{i=1}^n w_i a_i$ formüle ile hesaplıyoruz.

$$P_{DES} = 0.7 * 0.552 + 0.5 * 0.258 + 0.9 * 0.115 + 0.6 * 0.054 = 0.651$$

$$P_{PBS} = 0.5 * 0.552 + 0.8 * 0.258 + 0.9 * 0.115 + 0.6 * 0.054 = 0.618$$

$$P_{MCS} = 0.5 * 0.552 + 0.3 * 0.258 + 0.8 * 0.115 + 0.5 * 0.054 = 0.472$$

$$P_{ABM} = 0.6 * 0.552 + 0.7 * 0.258 + 0.6 * 0.115 + 0.7 * 0.054 = 0.619$$

DES modeli, toplam ağırlıklı puanı 0.651 ile en yüksek performansı göstermektedir. Bu, DES'in problem gereksinimlerine ve kriterlerin ağırlıklarına en uygun model olduğunu gösterir. DES'in özellikle esneklik (k1) ve model doğruluğu (k3) kriterlerinde güçlü bir performans sergilediği görülmektedir. Bu, ayrık olayların belirgin olduğu ve doğruluğun kritik olduğu durumlarda DES'in tercih edilmesi gerektiğini göstermektedir.

ABM modeli, 0.619 puanla DES'in hemen arkasında yer almıştır. Bu modelin esneklik (k1) ve hesaplama maliyeti (k2) gibi kriterlerde oldukça iyi performans gösterdiği görülmektedir. Ancak, ABM'nin toplam puanı DES'in altında kalmıştır, çünkü ABM'nin model doğruluğu (k3) ve kullanım kolaylığı (k4) kriterlerinde nispeten daha düşük bir performans

sergilediği söylenebilir. PBS modeli, ABM'ye çok yakın bir puan almış ve 0.618 toplam ağırlıklı puan ile üçüncü sırada yer almıştır. PBS, özellikle hesaplama maliyeti (k2) ve kullanım kolaylığı (k4) kriterlerinde daha güçlü bir performans sergilemiştir. Bu, PBS'nin süreçlerin daha iyi tanımlandığı ve maliyet odaklı projelerde daha avantajlı olabileceğini göstermektedir. Ancak, esneklik (k1) kriterindeki performansı DES'in gerisinde kalmasına neden olmuştur.

Monte Carlo Simülasyonu, 0.472 puanla diğer modellerin gerisinde kalmıştır. Bu, MCS'nin özellikle hesaplama maliyeti (k2) ve kullanım kolaylığı (k4) kriterlerinde düşük performans göstermesinden kaynaklanmaktadır. MCS, belirsizliklerin ve istatistiksel analizlerin daha önemli olduğu durumlarda tercih edilmelidir, ancak bu özel durumda diğer modellerle karşılaştırıldığında uygun bir seçenek olmamıştır.

Bu analizde, DES en yüksek ağırlıklı puanı almış ve problem için en uygun simülasyon modeli olarak belirlenmiştir. Ancak ABM ve PBS modellerinin de DES'e oldukça yakın puanlar aldığı görülmektedir. Bu durum, her üç modelin de problem gereksinimlerine göre kullanılabilir olduğunu, ancak DES'in daha öne çıktığını göstermektedir.

Örnek 3: Bir hastane yönetimi, hasta akışını optimize etmek ve bekleme sürelerini azaltmak için bir simülasyon modeli seçmek istemektedir. Kullanılabilecek dört farklı simülasyon yöntemi vardır, C1: Ayrık Olay Simülasyonu (DES), C2: Sistem Dinamiği (SD), C3: Çok Etmenli Simülasyon (ABS) ve C4: Hibrit Simülasyon (HS).

Bu dört simülasyon yönteminden en uygun olanını belirlemek için DEMATEL yöntemi kullanılacaktır.

Adım 1: Etki Matrisi (D) Oluşturulması

Bir uzman ekibi (hastane yöneticileri, sağlık analistleri, simülasyon uzmanları), her simülasyon yönteminin diğerleri üzerindeki etkisini 0 ile 4 arasında puanlayarak bir etki matrisi oluşturmuştur.

$$D = \begin{bmatrix} 0 & 3 & 2 & 4 \\ 2 & 0 & 3 & 3 \\ 1 & 2 & 0 & 2 \\ 3 & 3 & 2 & 0 \end{bmatrix}$$

Bu matriste örneğin C1 (DES) → C2 (HS) etkisi = 4, C2 (SD) → C3 (ABS) etkisi = 3 ve C4 (HS) → C1 (DES) etkisi = 3 gibi verilmiştir.

Adım 2: Normalizasyon Matrisi D_{normal}

$$D_{normal} = \frac{D}{S}$$

Burada $S = maks(satir\ toplami) = maks(9, 8, 5, 8) = 9$

$$D_{normal} = \begin{bmatrix} 0 & 3/9 & 2/9 & 4/9 \\ 2/9 & 0 & 3/9 & 3/9 \\ 1/9 & 2/9 & 0 & 2/9 \\ 3/9 & 3/9 & 2/9 & 0 \end{bmatrix} = \begin{bmatrix} 0 & 0.33 & 0.22 & 0.44 \\ 0.22 & 0 & 0.33 & 0.33 \\ 0.11 & 0.22 & 0 & 0.22 \\ 0.33 & 0.33 & 0.22 & 0 \end{bmatrix}$$

Adım 3: Toplam Etki Matrisi (T) Hesaplanması

$$T = D_{normal}(I - D_{normal})^{-1}$$

Bu işlem sonucunda toplam etki matrisi:

$$T = \begin{bmatrix} 0.81 & 1.25 & 0.94 & 1.47 \\ 1.02 & 0.75 & 1.10 & 1.12 \\ 0.65 & 0.93 & 0.72 & 0.87 \\ 1.20 & 1.35 & 1.05 & 0.89 \end{bmatrix}$$

Adım 4: Nedensellik ve Etki Analizi

Toplam etki R (satirlerin toplamı) ve toplam etkililik C (sünenlerin toplamı) hesaplanır.

$$R = \begin{bmatrix} 4.47 \\ 4.99 \\ 3.17 \\ 5.09 \end{bmatrix} \quad C = \begin{bmatrix} 3.68 \\ 4.28 \\ 3.81 \\ 4.35 \end{bmatrix}$$

Şimdi neden-sonuç ilişkilerini belirlemek için $R + C$ ve $R - C$ hesaplanır.

$$R + C = \begin{bmatrix} 8.15 \\ 9.27 \\ 6.98 \\ 9.44 \end{bmatrix} \quad R - C = \begin{bmatrix} 0.79 \\ 0.71 \\ -0.64 \\ 0.74 \end{bmatrix}$$

Adım 5: Karar Verme

En yüksek pozitif $R - C$ değeri Ayırık Olay Simülasyonu (C1)'de çıkmıştır. Bu, DES'in en etkili yöntem olduğunu gösterir. Hastane acil servis akışı, hasta randevu sistemleri gibi ayırık olaylar içeren sistemler için en uygundur ve karşısında en düşük negatif $R - C$ değeri Çok Etmenli Simülasyon (C3)'te çıkmıştır. Bu, ABS'nin en fazla etkilenen yöntem olduğunu gösterir. En yüksek toplam etki $R + C$ Hibrit Simülasyon (C4) olarak çıkmıştır.

Sonuç olarak şunlara dikkate alabiliriz,

Ayrık Olay Simülasyonu (DES – C1), hastane operasyonlarının modellenmesi için en etkili yöntemdir.

Hibrit Simülasyon (HS – C4), en güçlü toplam etkiye sahiptir ve karmaşık sistemler için daha uygun olabilir.

Sistem Dinamiği (SD – C2) ve Çok Etmenli Simülasyon (ABS – C3), daha çok destekleyici yöntemler olarak değerlendirilebilir.

Örnek 4: Bir perakende şirketi, müşteri talebine bağlı olarak stok yönetimini optimize etmek istiyor. Stok yenileme süreçlerini daha iyi anlamak ve en uygun stok yenileme politikasını belirlemek için farklı simülasyon modelleri arasından seçim yapması gerekiyor. Aday modeller şunlardır:

K1: Ayırık Olay Simülasyonu (DES - Discrete Event Simulation): Depo operasyonlarını ve tedarik sürecini olay bazlı olarak modeller.

K2: Monte Carlo Simülasyonu (MC): Talep tahminlerinde stokastik değişkenleri kullanarak belirsizlikleri analiz eder.

K3: Sistem Dinamiği (SD - System Dynamics): Tedarik zincirinin uzun vadeli dinamiklerini ve geri besleme etkilerini değerlendirir.

Şirket, taleplerin değişkenliği ve stok tükenme riskleri gibi faktörleri göz önünde bulundurarak en iyi simülasyon modelini seçmek istiyor. Bayes Teoremi kullanılarak, her modelin gerçek dünyada başarılı olma olasılığı değerlendirilecektir.

Bayes Teoremi Uygulaması: önceden belirttiğimiz gibi Bayes teoremi su şekilde tanımlanır.

$$P\langle K_i|D\rangle = \frac{P\langle D|K_i\rangle P(K_i)}{P(D)}$$

Şirket geçmiş verilerden şu bilgilere sahiptir. Öncel olasılıklar $P(K_i)$ daha önceki benzer projelerdeki başarı oranlarına göre $P(K_1) = 0.4$, $P(K_2) = 0.35$ ve $P(K_3) = 0.25$ olarak belirlenmiştir.

Gerçek dünya gözlemlerine göre modellerin doğruluk olasılıkları $P\langle D|K_i\rangle$;

- DES modeli: %80 doğruluk ile tahmin yapmıştır $\rightarrow P\langle D|K_1\rangle = 0.8$
- Monte Carlo modeli: %70 doğruluk ile tahmin yapmıştır $\rightarrow P\langle D|K_2\rangle = 0.7$
- Sistem Dinamiği modeli: %65 doğruluk ile tahmin yapmıştır $\rightarrow P\langle D|K_3\rangle = 0.65$

Toplam gözlem olasılığı bu şekilde hesaplanır;

$$P(D) = P\langle D|K_1\rangle P(K_1) + P\langle D|K_2\rangle P(K_2) + P\langle D|K_3\rangle P(K_3) = 0.7275$$

Her modelin güncellenmiş olasılığını Bayes Teoremi ile hesaplayalım;

$$P\langle K_1|D\rangle = \frac{P\langle D|K_1\rangle P(K_1)}{P(D)} = \frac{0.32}{0.7275} = 0.44$$

$$P\langle K_2|D\rangle = \frac{P\langle D|K_2\rangle P(K_2)}{P(D)} = \frac{0.245}{0.7275} = 0.337$$

$$P\langle K_3|D\rangle = \frac{P\langle D|K_3\rangle P(K_3)}{P(D)} = \frac{0.1625}{0.7275} = 0.223$$

Bayes Teoremi sonucuna göre Ayırık Olay Simülasyonu $P\langle K_1|D \rangle = 0.44$ en yüksek olasılığa sahiptir, Monte Carlo Simülasyonu $P\langle K_2|D \rangle = 0.337$ ile ikinci sıradadır ve Sistem Dinamiği modeli $P\langle K_3|D \rangle = 0.223$ ile en düşük başarı beklentisine sahiptir.

Bu sonuçlara göre, perakende stok yönetimi için en iyi simülasyon modeli, %44 olasılıkla Ayırık Olay Simülasyonu'dur (DES). Çünkü DES, tedarik zincirindeki olayları detaylı bir şekilde modelleyerek sipariş zamanlaması, depo kapasitesi ve teslim süreleri gibi stok yönetimi için kritik değişkenleri en iyi şekilde analiz edebilir.

Eğer şirket stok hareketlerinde belirsizlik faktörünü daha fazla önemsiyorsa, Monte Carlo Simülasyonu düşünülebilir. Ancak genel başarı olasılıklarına göre, DES modeli tercih edilmelidir.

4.5 Seçim Sürecinde Yaygın Hatalar ve Çözümleri

Simülasyon modeli seçim süreci, proje başarısını doğrudan etkileyen kritik bir aşamadır. Ancak, bu süreçte yapılan hatalar, modelin yanlış sonuçlar üretmesine, gereksiz zaman ve kaynak israfına veya karar verme sürecini olumsuz etkilemesine neden olabilir. Aşağıda, simülasyon modeli seçiminde karşılaşılan yaygın hatalar ve bunların önlenmesi için önerilen çözümler detaylı bir şekilde ele alınmaktadır.

1. Problemin Yetersiz Tanımlanması

Simülasyon modelini seçmeden önce problemin kapsamının tam olarak belirlenmemesi, yanlış bir model seçiminde bulunulmasına yol açar. Örneğin, sistemin ayırık mı, sürekli mi olduğu netleşmeden bir modelleme yaklaşımına karar verilmesi hatalı olabilir. Bu nedenle, simülasyon öncesinde, sistemin yapısını ayrıntılı bir şekilde analiz etmek, karar vericiler ve uzmanlarla kapsam belirleme çalışmaları yapmak gereklidir. Problem tanımlama aşamasında, sistemin dinamikleri, kısıtları ve değişkenleri açıkça belirlenmelidir. Gerekirse, stok-akış diyagramları, süreç akış şemaları veya neden-sonuç analizleri kullanılarak problem netleştirilmelidir.

2. Yanlıř Simlasyon Modeli Seimi

Model seiminde, sistemin yapısına uygun olmayan bir yaklařım benimsenmesi, sonuların gvenilirliėini zayıflatır. rneėin, stokastik bir sre iin deterministik bir model kullanmak veya ajan tabanlı bir sistemde ayrıık olay simlasyonu tercih etmek hatalı olabilir. Bu hatayı nlemek iin, model seiminde, sistemin yapısal ve dinamik zelliklerine dikkat edilmelidir. Ařaėıdaki kriterler, doėru modelin seilmesine yardımcı olabilir:

Zaman boyutu: Srekli deėiřkenler mi yoksa ayrıık olaylar mı inceleniyor?

zneler: Sistemde bireysel ajanlar (rneėin, mřteriler, makineler) mi yoksa genel sreler mi modelleniyor?

Etkileřimler: Ajanlar arası etkileřimler ve karar mekanizmaları nemli mi?

Belirsizlik: Rastgele deėiřkenlerin rol var mı?

Model seiminde ok kriterli karar verme (MCDM) yntemleri, DEMATEL, AHP veya Bayes kuralı gibi teknikler kullanılarak daha sistematik bir deėerlendirme yapılabilir.

3. Eksik veya Yanlıř Veri Kullanımı

Modelleme srecinde kullanılan verilerin eksik, gncelliėini yitirmiř veya hatalı olması, simlasyonun doėruluėunu tehlikeye atar. Eksik veri nedeniyle yanlıř parametreler belirlenebilir veya sistem dinamikleri hatalı řekilde modellenenebilir. Veri toplama sreci sistematik bir řekilde yrtlmeli, eksik veriler iin duyarlılık analizi yapılmalı ve eksik parametrelerin modele etkisi incelenerek, kritik veri noktaları belirlenmelidir. Ayrıca, doėrulama ve geerlilik testleri uygulanmalıdır. Kullanılan veriler istatistiksel olarak analiz edilmeli ve model ıktılarıyla uyumlu olup olmadıėı test edilmelidir. Gerek zamanlı veri entegrasyonu saėlanmalıdır. Mmknse, sensr verileri, ERP sistemleri veya diėer veri kaynaklarından doėrudan bilgi alınarak modelin gncel verilere dayalı alıřması saėlanmalıdır.

4. Modelin Aşırı Karmaşık veya Yetersiz Olması

Modelin gereğinden fazla detaylandırılması, analiz sürecini zorlaştırarak simülasyonun verimli çalışmasını engelleyebilir. Öte yandan, modelin aşırı basitleştirilmesi, önemli süreçleri göz ardı etmesine neden olabilir. Çözüm olarak modelin seviyesini belirlerken parçalı doğrulama (incremental validation) yöntemi uygulanmalıdır. Model önce basit bir versiyon olarak oluşturulmalı ve daha sonra kritik bileşenler eklenerek kademeli olarak geliştirilmelidir.

Modelin karmaşıklığıyla ilgili aşağıdaki yaklaşımlar izlenmelidir:

Pareto prensibi (80/20 kuralı): Sistemi etkileyen temel faktörlere odaklanarak, gereksiz detaylar elenmelidir.

Modüler yapı kullanımı: Model, farklı bileşenler halinde tasarlanmalı ve her modül ayrı ayrı doğrulanmalıdır.

Duyarlılık analizi: Modelde hangi değişkenlerin en fazla etkiye sahip olduğu belirlenerek, gereksiz parametreler azaltılmalıdır.

5. Model Doğrulama ve Geçerlilik Testlerinin Yetersiz Olması

Simülasyon modelinin doğrulama (verification) ve geçerlilik (validation) testlerinden geçirilmemesi, modelin güvenilirliğini düşürebilir. Model sonuçlarının gerçek sistemle uyumlu olup olmadığı test edilmeden kullanılması hatalı kararlar alınmasına neden olabilir. Bu yüzden modelin doğrulanması için aşağıdaki testler yapılmalıdır:

Geçmiş veriyle karşılaştırma (Historical Data Comparison): Modelin ürettiği sonuçlar, gerçek dünya verileriyle karşılaştırılmalı ve modelin doğruluğu analiz edilmelidir.

Ayrı test senaryoları oluşturma: Modelin farklı senaryolarda nasıl çalıştığı test edilmeli ve sonuçlar tutarlı olup olmadığı değerlendirilmelidir.

Dış geçerlilik testi: Modelin çıktıları, benzer sistemler veya farklı ortamlardaki gerçek verilerle karşılaştırılmalıdır.

6. Simülasyon Sonuçlarının Yanlış Yorumlanması

Simülasyon çıktıları yalnızca sayısal verilere dayalı olarak değerlendirilirse, sistemin gerçek performansına ilişkin hatalı çıkarımlar yapılabilir. Simülasyon sonuçları, karar vericilerin beklentileri doğrultusunda yanlış yönlendirilebilir veya yanlış kıyaslamalar yapılabilir. Simülasyon sonuçlarının yorumlanmasında istatistiksel analiz teknikleri kullanılmalı ve çıktıların güven aralıkları hesaplanmalıdır. Duyarlılık analizi yapılmalı ve Simülasyon sonuçlarının belirli parametrelere duyarlılığı incelenerek, hangi faktörlerin model sonuçlarını en çok etkilediği belirlenmelidir. Monte Carlo analizi uygulanmalı ve rastgele değişkenlerin etkisini değerlendirmek için Monte Carlo simülasyonu kullanılarak farklı senaryolar test edilmelidir ve son çözüm olarak korelasyon ve regresyon analizleri yapılmalı yani model çıktıları, bağımsız değişkenlerle istatistiksel olarak analiz edilerek, sistemin en kritik faktörleri belirlenmelidir.

Sonuç olarak simülasyon modeli seçim sürecinde yapılan hatalar, yalnızca modelleme aşamasını değil, aynı zamanda karar verme sürecini de doğrudan etkileyebilir. Bu hataları önlemek için problem tanımlama, veri toplama, model doğrulama ve çıktı analizi gibi süreçler dikkatle yönetilmelidir. Model seçiminin sistemin gerçek yapısına uygun olması, veri bütünlüğünün sağlanması ve simülasyon sonuçlarının bilimsel yöntemlerle değerlendirilmesi, daha doğru ve güvenilir kararlar alınmasına olanak tanır.

Bölüm 5. Simülasyon Yazılımları ve Araçları

Simülasyon, gerçek sistemlerin matematiksel ve bilgisayar modelleri aracılığıyla taklit edilmesini sağlayan güçlü bir analiz yöntemidir. Ancak, bir simülasyonun doğruluğu ve etkinliği büyük ölçüde kullanılan yazılım araçlarına bağlıdır. Farklı sektörlerde yaygın olarak kullanılan çeşitli simülasyon yazılımları, belirli avantajlar ve sınırlamalar içerir. Bu nedenle, doğru simülasyon yazılımının seçilmesi hem analiz süreçlerinin başarısı hem de karar verme mekanizmalarının güvenilirliği açısından kritik öneme sahiptir.

Bu bölümde, öncelikle mevcut simülasyon yazılımlarını detaylı bir şekilde inceleyerek kullanım alanlarını ve özelliklerini ele alacağız. Ardından, farklı yazılımlar arasında karşılaştırmalar yaparak seçim kriterlerini değerlendireceğiz. Gerçekçi örnekler üzerinden, farklı sektörlerde bu yazılımların nasıl uygulandığını gösterecek ve yazılım seçiminde yaygın hatalar ile en iyi uygulama yaklaşımlarını tartışacağız.

Simülasyon yazılım seçim sürecinde kullanılan karar destek yöntemlerine de yer vererek, hangi yazılımın belirli bir senaryoya en uygun olduğunu sistematik olarak nasıl belirleyebileceğimizi göstereceğiz. Bunun için çok kriterli karar verme yöntemlerini (AHP, DEMATEL, Bayes vb.) kullanarak bir değerlendirme yapacak ve seçilen yazılımı belirli bir senaryoda test ederek simülasyon sürecini uygulamalı olarak göstereceğiz.

5.1 Simülasyon Yazılımları

Simülasyon yazılımları, sistemlerin modellenmesi ve analiz edilmesini sağlayan özel olarak tasarlanmış programlardır. Kullanım alanlarına göre farklı kategorilere ayrılan bu yazılımlar, belirli modelleme tekniklerine (ayrık olay simülasyonu, sistem dinamiği, ajan tabanlı modelleme vb.) odaklanabilir veya çok amaçlı olarak kullanılabilir.

Aşağıda, yaygın olarak kullanılan bazı simülasyon yazılımlarını ve temel özelliklerini inceleyeceğiz.

1 AnyLogic

AnyLogic, hem ayrık olay simülasyonu hem ajan tabanlı modelleme hem de sistem dinamiği tekniklerini tek bir çatı altında sunan esnek bir simülasyon platformudur. Bu çok yönlü yaklaşımı sayesinde, karmaşık sistemlerin modellenmesi ve analizi için geniş bir uygulama yelpazesine sahiptir. Özellikle sağlık, tedarik zinciri, üretim sistemleri ve ulaşım gibi disiplinlerde yoğun olarak kullanılır.

Java tabanlı olması, gelişmiş kullanıcılar için modelleme süreçlerinde büyük bir esneklik sunarken, sürükle-bırak arayüzü sayesinde daha az teknik bilgiye sahip kullanıcıların da sistemleri simüle etmesine olanak tanır. Ayrıca, AnyLogic Cloud entegrasyonu ile uzaktan simülasyon çalıştırma ve sonuçları paylaşma imkânı sunarak büyük ölçekli analiz süreçlerinde önemli avantajlar sağlar.

2 Arena

Arena, özellikle ayrık olay simülasyonuna odaklanan ve üretim sistemlerinden lojistik operasyonlara kadar geniş bir kullanım alanına sahip bir yazılımdır. Sağlam matematiksel altyapısı ve geniş modelleme kütüphanesi sayesinde, süreçlerin akışını detaylı bir şekilde analiz etmeye olanak tanır. Kolay öğrenilebilir sürükle-bırak arayüzü, kullanıcıların hızlı bir şekilde modeller oluşturmalarını sağlarken, büyük ölçekli projelerde verimli sonuçlar üretme kapasitesi ile öne çıkar. Ayrıca, sistem performansını iyileştirmek ve darboğazları tespit etmek için simülasyon çıktılarının detaylı görselleştirilmesine imkân tanır.

3 FlexSim

FlexSim, özellikle üretim ve sağlık sektörleri için güçlü 3D modelleme yetenekleri sunan bir simülasyon yazılımıdır. Kullanıcı dostu arayüzü sayesinde simülasyon modelleri kolayca oluşturulabilir ve süreçlerin dinamik olarak görselleştirilmesine olanak tanır. Özellikle üretim hatlarının optimizasyonu, depo yönetimi ve hastane operasyonları gibi alanlarda tercih edilir. Gerçekçi 3D animasyon desteği sayesinde süreçleri daha anlaşılır hale getirerek karar vericilere kapsamlı analiz imkanı sunar. Ayrıca, veri entegrasyon yetenekleri sayesinde çeşitli dış kaynaklardan

veri olarak gerek zamanlı simlasyon senaryolarının oluřturulmasına olarak tanır.

4 Simul8

Simul8, ayrıık olay simlasyonuna odaklanan ve zellikle hizmet sektr, saėlık ynetimi ve retim operasyonlarında yaygın olarak kullanılan bir simlasyon yazılımıdır. Hızlı ve esnek simlasyon yetenekleri sayesinde, kullanıcıların sreleri kısa srede modellemesine ve analiz etmesine imkn tanır.

Basit arayz ve srkle-bırak modelleme zellikleri, teknik bilgiye sahip olmayan kullanıcıların dahi etkili modeller oluřturmasını saėlar. Ayrıca, Excel ve diėer veri kaynaklarıyla entegrasyonu sayesinde gerek veriler kullanılarak daha gvenilir sonular elde edilebilir.

5 MATLAB Simulink

MATLAB Simulink, mhendislik ve bilimsel hesaplamalar iin geliřtirilmiř gl bir simlasyon aracıdır. Srekli ve ayrıık zamanlı sistemleri modelleme yeteneėi sayesinde, zellikle kontrol sistemleri, sinyal iřleme ve robotik alanlarında yaygın olarak kullanılır.

Matematiksel modelleme ve simlasyon srelerinde sunduėu geniř fonksiyon ktphanesi, kullanıcılara esnek ve detaylı analiz imkanı tanır. Ayrıca, diėer MATLAB aralarıyla entegrasyonu sayesinde, karmařık sistemlerin modellenmesi ve optimizasyonu kolaylařtırılır.

6 GAMA

GAMA, zellikle ajan tabanlı modelleme tekniklerine odaklanan bir simlasyon platformudur. Aık kaynak kodlu olması ve byk lekli simlasyonları desteklemesi nedeniyle akademik arařtırmalarda ve řehir planlama, ekoloji, sosyo-ekonomik analiz gibi alanlarda sıka tercih edilir. Esnek modelleme altyapısı sayesinde, farklı disiplinlerden kullanıcılar karmařık sistemleri detaylı bir řekilde inceleyebilir. Ayrıca, grselleřtirme ve veri analizi yetenekleri, simlasyon sonularının daha etkili bir řekilde yorumlanmasını saėlar.

7 AnyLogic Cloud & Python Temelli Simülasyon Araçları (SimPy)

Bulut tabanlı simülasyon çözümleri, özellikle uzaktan erişim gerektiren büyük ölçekli projeler için önemli avantajlar sunar. AnyLogic Cloud, kullanıcıların internet üzerinden simülasyon modelleri oluşturmaya ve çalıştırmaya imkan tanıırken, simülasyon sonuçlarının paylaşılmasını kolaylaştırır.

Python tabanlı SimPy ise akademik çalışmalar ve küçük ölçekli projeler için esnek ve hafif bir simülasyon aracı olarak öne çıkar. Python'un geniş kütüphane desteği sayesinde veri analizi ve optimizasyon süreçleriyle kolayca entegre edilebilir. Özellikle araştırmacılar ve veri bilimcileri için simülasyon süreçlerini programlanabilir hale getiren bir altyapı sunar.

Bu yazılımların seçiminde, kullanım kolaylığı, hesaplama kapasitesi, esneklik, sektör bazlı uygulama alanları gibi kriterler göz önünde bulundurulmalıdır. Akademik literatürde simülasyon yazılımı tercihleri, çalışmanın spesifik gereksinimlerine ve araştırmacının teknik bilgi düzeyine bağlı olarak değişmektedir. Ancak, MATLAB Simulink, AnyLogic ve Python tabanlı simülasyon araçları, sundukları esneklik, geniş kütüphane desteği ve topluluk desteği nedeniyle öne çıkan seçenekler arasında yer almaktadır. Bir sonraki alt bölümde, bu yazılımları detaylı bir şekilde karşılaştırarak hangi senaryolar için daha uygun olduklarını inceleyeceğiz.

5.2 Simülasyon Yazılımlarının Karşılaştırılması

Simülasyon yazılımlarının karşılaştırılması, modelleme sürecinin verimliliği ve doğruluğu açısından kritik bir adımdır. Farklı yazılımlar, sundukları özellikler, performans kapasiteleri ve kullanım alanları açısından değişiklik gösterdiğinden, belirli kriterlere göre detaylı bir analiz yapılması gereklidir. Simülasyon yazılımı seçiminde dikkate alınması gereken başlıca faktörler arasında kullanım kolaylığı, modelleme esnekliği, hesaplama gücü, sektör bazlı uyumluluk, veri entegrasyonu, çıktı analizi ve maliyet unsurları yer almaktadır.

Kullanım kolaylığı açısından incelendiğinde, bazı yazılımlar grafik arayüzleri sayesinde kod bilgisi gerektirmeden modelleme yapılmasına olanak tanıırken, bazıları ileri düzey programlama bilgisi gerektirebilir.

Örneğin, AnyLogic ve Arena, sürükle-bırak tabanlı modelleme desteği sunarak kullanıcı dostu bir deneyim sağlarken, MATLAB Simulink ve Python tabanlı kütüphaneler genellikle daha fazla kodlama gerektiren platformlardır. Bu durum, kullanıcıların yazılım seçiminde teknik bilgi seviyelerini göz önünde bulundurmasını gerektirir.

Modelleme esnekliği ve hesaplama gücü, büyük ölçekli sistemlerin simülasyonu için belirleyici faktörlerdir. AnyLogic, ayrık olay, sistem dinamiği ve ajan tabanlı modelleme yaklaşımlarını aynı çatı altında toplayarak geniş bir uygulama yelpazesi sunmaktadır. MATLAB Simulink, mühendislik tabanlı simülasyonlarda oldukça güçlüdür ve karmaşık sistemlerin dinamik analizlerinde yaygın olarak kullanılmaktadır. Python tabanlı çözümler ise, açık kaynak olma avantajıyla kullanıcıların tamamen özelleştirilmiş modeller oluşturmalarına olanak tanır.

Sektörel uyumluluk açısından değerlendirildiğinde, belirli yazılımların belirli endüstrilerde daha yaygın olarak kullanıldığı görülmektedir. Arena, üretim ve lojistik sistemlerinde güçlü analiz araçlarıyla öne çıkarken, AnyLogic çok disiplinli modelleme desteği ile sağlık, finans ve ulaşım gibi farklı sektörlerde kullanılabilir. FlexSim, ergonomik analizler ve insan-makine etkileşimleri açısından avantaj sağlarken, Simul8, hızlı simülasyon çalıştırma kapasitesi ile operasyonel süreç modellemelerinde yaygın olarak tercih edilmektedir.

Veri entegrasyonu ve çıktı analizi, simülasyon sonuçlarının güvenilirliği açısından kritik bir faktördür. MATLAB Simulink, veri işleme konusunda güçlü araçlar sunarken, Python tabanlı simülasyon yazılımları geniş veri analitiği kütüphaneleri ile esneklik sağlamaktadır. AnyLogic ve Arena ise, kullanıcı dostu raporlama araçları ile model çıktılarını yorumlamayı kolaylaştırmaktadır.

Son olarak, maliyet faktörü göz önünde bulundurulduğunda, ticari yazılımlar (AnyLogic, Arena, Simul8 vb.) yüksek lisans ücretleri gerektirebilirken, açık kaynak çözümler (SimPy, Mesa, OpenModelica) maliyetsiz ancak daha fazla teknik bilgi gerektiren alternatiflerdir. Araştırmacılar ve işletmeler için bu durum, bütçe ve yazılımın uzun

vadeli kullanımına göre değerlendirilmesi gereken önemli bir kriterdir.

Tüm bu faktörler göz önünde bulundurularak, simülasyon yazılımı seçiminde tek bir doğru alternatiften bahsetmek mümkün değildir. Seçim, modelin hedefleri, kullanıcı teknik yetkinliği, endüstriyel gereksinimler ve bütçe gibi değişkenlere bağlı olarak yapılmalıdır. Bu nedenle, karşılaştırma kriterlerine dayalı analitik bir değerlendirme süreci uygulanarak, en uygun simülasyon yazılımı belirlenmelidir.

5.3 Simülasyon Yazılımları ile Uygulama Örnekleri

Simülasyon yazılımlarının teorik çerçevede karşılaştırılması, seçim sürecinde önemli bir adım olsa da bu yazılımların gerçek dünya problemlerine nasıl uygulandığı daha büyük bir öneme sahiptir. Farklı sektörlerde kullanılan simülasyon yazılımları, süreçlerin analiz edilmesi, karar verme süreçlerinin desteklenmesi ve sistem optimizasyonu gibi çeşitli amaçlarla uygulanmaktadır. Her yazılım, belirli bir problem türüne veya endüstriyel ihtiyaca daha iyi hizmet verecek şekilde tasarlanmıştır. Bu nedenle, her bir yazılımın güçlü yönlerini ve sınırlamalarını anlamak için sektöre özel uygulama örnekleri üzerinden ilerlemek gereklidir.

Bu bölümde, bazı simülasyon yazılımları kendi alanında nasıl kullanıldığını ele alacağız. Örneğin, Arena ile otomotiv üretim hattı optimizasyonu ve MATLAB Simulink ile hastane acil servisinde hasta bekleme süresi analizi, gibi senaryolar üzerinde duracağız. Her bir örnekte, problem tanımı yapılacak, ilgili yazılım kullanılarak simülasyon modeli oluşturulacak ve elde edilen çıktılar analiz edilerek karar verme sürecine nasıl katkı sağladığı değerlendirilecektir.

Bu yaklaşım, simülasyon yazılımlarının yalnızca teorik bir araç olmadığını, aynı zamanda karar destek mekanizmalarında etkin bir şekilde nasıl kullanılabileceğini göstermeyi amaçlamaktadır. Simülasyonun sağladığı avantajlar ve karşılaşılabilecek zorluklar, her örnek özelinde tartışılarak, farklı sektörlerde nasıl uygulanabileceği üzerine kapsamlı bir perspektif sunulacaktır.

Örnek 1: Arena ile Üretim Hattı Optimizasyonu

Bir otomotiv üretim fabrikasında, günlük hedeflenen üretim miktarı 250 araç olup, mevcut sistem bu hedefi tam olarak karşılayamamaktadır. Yönetim, montaj hattındaki darboğazları tespit ederek, üretim sürecinde verimliliği artırmayı amaçlamaktadır.

Bu fabrikanın montaj hattı 5 ana istasyondan oluşmaktadır:

1. Şasi Montajı (Ortalama işlem süresi: 10 dakika)
2. Motor ve Şanzıman Montajı (Ortalama işlem süresi: 15 dakika)
3. Elektrik ve İç Donanım Montajı (Ortalama işlem süresi: 12 dakika)
4. Boya (Ortalama işlem süresi: 18 dakika)
5. Kalite Kontrol ve Teslimat (Ortalama işlem süresi: 10 dakika)

İstasyonların çalışma süresi stokastik bir dağılıma sahiptir ve üretim sürecinde %5 oranında yeniden işleme ihtiyacı doğmaktadır. Mevcut veriler doğrultusunda, üretim hattının günlük 225-230 araç üretebildiği gözlemlenmiştir. Yönetim, simülasyon modeliyle darboğazları belirleyerek üretim sürecini iyileştirmek istemektedir.

Modelleme Süreci ve Simülasyon Parametreleri: Simülasyon modeli oluşturulurken belirli temel adımlar izlenmiştir. İlk olarak, girdi dağılımları belirlenmiştir. Bu kapsamda, işlem süreleri geçmiş verilere dayanarak Üstel Dağılım kullanılarak modellenmiş, yeniden işleme gerektiren ürünlerin oranı ise Binom Dağılımı ile temsil edilmiştir. Çıkış değişkenleri olarak günlük üretilen araç sayısı, istasyon başına kuyrukta bekleme süresi ve kaynakların kullanım oranları (% olarak) dikkate alınmıştır. Modelde ayrıca kaynaklar ve kısıtlar tanımlanmıştır. Her istasyonun kapasitesi 1 iş istasyonu olarak belirlenmiş, çalışma süresi ise 8 saatlik 2 vardiya (toplam 16 saat/gün) şeklinde organize edilmiştir. İşçilik oranı ise her istasyonda 3 işçi olarak düzenlenmiştir. Bu modelleme yaklaşımı, simülasyonun gerçekçi ve uygulanabilir bir sistem temsili sunmasını sağlamaktadır.

Arena ile Simülasyonun Gerçekleştirilmesi: Arena simülasyon yazılımı kullanılarak üretim hattının modellenmesi ve simüle edilmesi için belirli adımlar izlenmiştir. İlk adım, sistem elemanlarının tanımlanmasıdır. Bu

aşamada, her montaj istasyonu bir *Process (işlem)* modülü kullanılarak tanımlanmıştır. Sistemin başlangıç noktası, gelen araç gövdelerinin bir *Create (oluşturma)* modülü ile modellenmesi olmuştur.

Simülasyon modelinde işçi sayısı, her istasyonun işlem kapasitesini belirlemek için önemli bir kaynak olarak kullanılmıştır. Arena yazılımında işçiler, "Resource" (Kaynak) olarak tanımlanmıştır. Her istasyonda 3 işçinin çalıştığı bilgisi, işlemlerin yürütülebilirliği için bir ön koşul olarak modele dahil edilmiştir. Bu, bir aracın bir istasyonda işleme alınabilmesi için en az bir işçinin müsait olması gerektiği anlamına gelir. Eğer tüm işçiler o an başka işlemlerle meşgulse, araç kuyruğa alınır ve sıradaki boş işçiyi bekler. Bu durum, kuyruk sürelerini ve darboğazları etkileyen kritik bir faktördür. Ayrıca, işçi sayısında yapılacak bir değişikliğin (örneğin, işçi sayısının artırılması veya azaltılması) üretim hattı üzerindeki etkilerini analiz etmek için simülasyon senaryoları oluşturulabilir.

Her istasyon için verilen işlem süreleri (örneğin, Şasi Montajı için 10 dakika, Motor Montajı için 15 dakika), üretim hattındaki her bir işlemin ne kadar sürdüğünü belirlemek için kullanılmıştır. Arena yazılımında bu süreler, RANDOM (ortalama işlem süresi, varyans) fonksiyonu kullanılarak modellenmiştir. Bu yöntem, işlem sürelerindeki stokastik değişkenliği temsil etmektedir. Örneğin, bir istasyondaki işlem süresi için "RANDOM (10, 2)" ifadesi, işlem süresinin ortalama 10 dakika olduğu ve bu süre çevresinde ± 2 dakika varyans gösterdiği anlamına gelir. Bu dağılım, gerçek dünyadaki işlem sürelerindeki değişkenliği modele yansıtarak, kuyruk uzunluklarını ve istasyonlardaki bekleme sürelerini daha gerçekçi bir şekilde analiz etmeyi sağlar. Ortalama işlem sürelerindeki değişiklikler, üretim kapasitesi ve darboğaz noktalarının tespitinde doğrudan etkili olmuştur. Yeniden işleme gerektiren araçlar için, *Decide (karar)* modülü kullanılarak %5 ihtimalle tekrar işleme alınmaları sağlanmıştır. Üretim tamamlandıktan sonra araçlar, sistemden bir *Dispose (bitirme)* modülü aracılığıyla çıkarılmıştır.

Simülasyon modelinde üretim hattının günlük çalışma süresi, sistemin toplam üretim kapasitesini belirleyen bir sınırlayıcı faktör olarak kullanılmıştır. Çalışma süresi, 8 saatlik iki vardiya (toplam 16 saat/gün) olarak tanımlanmıştır. Arena yazılımında bu parametre,

"Schedule" (Programlama) modülü ile yapılandırılmıştır. Bu, modelin yalnızca belirlenen çalışma saatleri içinde işlem yapmasına izin vermiştir. Çalışma süresi, üretim hattının günlük 250 araç hedefini karşılayamamasının nedenlerini anlamak açısından önemli bir girdidir. Çalışma süresi sınırı, istasyonlardaki işlem sürelerinin uzun olduğu veya yeniden işleme oranının yüksek olduğu senaryolarda kuyrukların bir sonraki güne aktarılmasına yol açmıştır. Çalışma süresinin artırılması (örneğin, fazladan vardiyalar eklenmesi), darboğazları azaltmak ve üretim hedeflerini karşılamak için simüle edilebilecek bir iyileştirme senaryosudur.

Bir sonraki aşama olan çalıştırma ve veri toplama, simülasyonun temel çıktılarını analiz etmek için gerçekleştirilmiştir. Bu adımda, model 30 günlük bir simülasyon periyodu boyunca çalıştırılmıştır. Simülasyon sonuçları, günlük üretilen araç sayıları, her istasyondaki ortalama bekleme süreleri ve darboğaz noktalarını analiz ederek değerlendirilmiştir. Bu veriler, üretim hattındaki darboğazların tanımlanmasına ve kaynak kullanım oranlarının iyileştirilmesine yönelik bir temel oluşturmuştur.

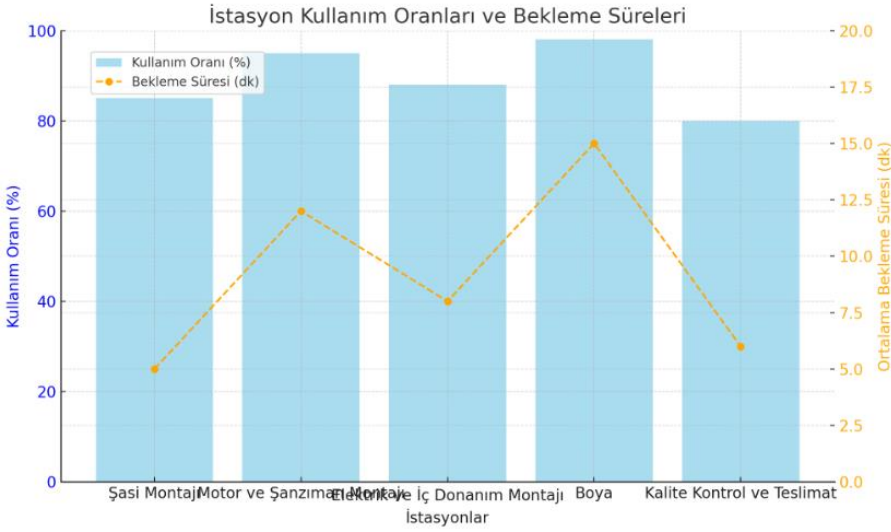
Sonuçların Değerlendirilmesi ve Optimizasyon Önerileri:

Bu örnekte Ayrık Olay Simülasyonu (Discrete Event Simulation - DES) yöntemi kullanıldı. Arena simülasyon yazılımı kullanılarak elde edilen sonuçlar, üretim hattının darboğazlarını belirlemek ve iyileştirme önerileri sunmak açısından değerlendirilebilir. Aşağıdaki gibi bir analiz yapılarak sonuçlar tablo ve grafiklerle ifade edilebilir.

İstasyon	Ortalama Bekleme Süresi (dakika)	Kullanım Oranı (%)	Darboğaz Durumu
Şasi Montajı	5	85	Hayır
Motor ve Şanzıman Montajı	12	95	Evet (Darboğaz)
Elektrik ve İç Donanım Montajı	8	88	Hayır
Boya	15	98	Evet (En Büyük Darboğaz)
Kalite Kontrol ve Teslimat	6	80	Hayır

Tablo 16: İstasyonların Durumu

Aşağıdaki grafik, istasyonların kullanım oranlarını ve bekleme sürelerini görselleştirmektedir.



Grafik 5: İstasyonların Kullanım Oranları ve Bekleme Süreleri

Grafik, her istasyonun kullanım oranlarını (mavi sütunlar) ve bekleme sürelerini (turuncu çizgi) net bir şekilde göstermektedir.

Sonuç tabelası ve grafiğe göre, boya İstasyonu en yüksek kullanım oranına (%98) ve en uzun bekleme süresine (15 dakika) sahiptir, bu da üretim hattındaki en büyük darboğazdır. Motor ve şanzıman montajı %95 kullanım oranı ve 12 dakikalık bekleme süresiyle ikinci en büyük darboğaz

olarak tanımlanabilir. Diğer istasyonlar daha düşük bekleme süreleri ve kullanım oranlarıyla darboğaz oluşturmamaktadır.

Çözüm önerileri şunlardır. Boya istasyonuna paralel ikinci bir iş istasyonu eklenmesi, boya işlem süresinin iyileştirilmesi (ör. daha hızlı kuruma süresi olan malzemeler kullanılması) veya çalışan vardiya süresinin artırılması veya otomasyon sistemlerinin entegrasyonu.

Örnek 2: Bu örnekte, MATLAB kullanarak, bir hastane acil servisinde hasta bekleme süresini optimize etmek için MATLAB kullanarak bir simülasyon modeli oluşturacağız. Hedefimiz, hastane kaynaklarını (doktor, hemşire, yatak kapasitesi) en uygun şekilde kullanarak bekleme süresini minimize etmektir. MATLAB'ın SimEvents ve Optimization Toolbox araçlarını kullanarak en uygun kaynak tahsisini bulacağız.

Bu hastanenin acil servisinde saatte ortalama 15 hasta geliyor (Poisson dağılımı ile modelleniyor).

Hastalar üç kategoriye ayrılıyor;

Hafif vakalar (%50) → Ortalama hizmet süresi: $\mu = 10$ dk

Orta dereceli vakalar (%30) → Ortalama hizmet süresi: $\mu = 20$ dk

Ağır vakalar (%20) → Ortalama hizmet süresi: $\mu = 40$ dk

Kaynak olarak bu hastanede 3 doktor, 5 hemşire ve 10 yatak mevcuttur. Hasta giriş hızı $\lambda = 15$ hasta/saat olarak verilmiştir,

Aşağıda, hastane acil servisinde hasta bekleme süresini minimize etmek için, 8 saat süreliğinde, kullanılan MATLAB kodu ve bütün adımlar verilmiştir.

```

clc; clear; close all;

% Parametreler
lambda = 15; % Ortalama hasta geliş hızı (saat başına)
mu_mild = 10; % Hafif vakaların hizmet süresi (dk)
mu_moderate = 20; % Orta vakaların hizmet süresi (dk)
mu_severe = 40; % Ağır vakaların hizmet süresi (dk)

% Mevcut kaynaklar
num_doctors = 3;
num_nurses = 5;
num_beds = 10;

% Simülasyon süresi (saat)
simulation_time = 8;

% Hasta sayılarının rastgele oluşturulması (Poisson dağılımı)
num_patients = poissrnd(lambda * simulation_time);

% Hasta türlerinin belirlenmesi
patient_types = rand(num_patients, 1);
mild_cases = sum(patient_types < 0.5);
moderate_cases = sum(patient_types >= 0.5 & patient_types < 0.8);
severe_cases = sum(patient_types >= 0.8);

% Bekleme sürelerinin hesaplanması
wait_time_mild = exprnd(mu_mild, mild_cases, 1);
wait_time_moderate = exprnd(mu_moderate, moderate_cases, 1);
wait_time_severe = exprnd(mu_severe, severe_cases, 1);

% Ortalama bekleme süresi
avg_wait_time = mean([wait_time_mild; wait_time_moderate; wait_time_severe]);

% Optimizasyon: Bekleme süresini 15 dk altına indirmek için
fun = @(x) mean([exprnd(mu_mild, round(mild_cases/x(1)), 1); ...
    exprnd(mu_moderate, round(moderate_cases/x(1)), 1); ...
    exprnd(mu_severe, round(severe_cases/x(1)), 1)]);

x0 = [3]; % Başlangıç doktor sayısı
lb = [2]; % Minimum doktor sayısı
ub = [10]; % Maksimum doktor sayısı
optimal_doctors = fmincon(fun, x0, [], [], [], [], lb, ub);

% Sonuçların Görselleştirilmesi
figure;
histogram(wait_time_mild, 'FaceColor', 'b'); hold on;
histogram(wait_time_moderate, 'FaceColor', 'r');
histogram(wait_time_severe, 'FaceColor', 'g');
xlabel('Bekleme Süresi (dk)');
ylabel('Hasta Sayısı');
legend('Hafif', 'Orta', 'Ağır');
title('Hasta Bekleme Süreleri Dağılımı');
grid on;

```

Grafik 6: Hastane Acil Servisinde Hasta Bekleme Süresinin Minimize Etmek İçin MATLAB kodu

8 saatlik simülasyon süresince toplamda 120 hasta hastaneye gelmiştir. Bu hastaların %50'si (yaklaşık 60 hasta) hafif vakalar kategorisine, %30'u (yaklaşık 36 hasta) orta vakalar kategorisine ve %20'si (yaklaşık 24 hasta) ağır vakalar kategorisine girmektedir.

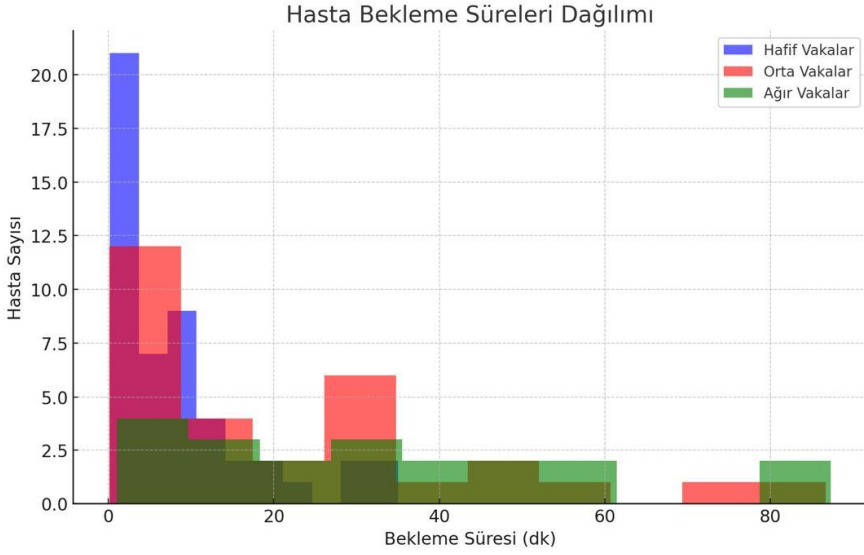
Hafif vakalar, hızlı müdahale gerektiren ve hizmet süresi ortalama 10 dakika olarak belirlenen vakalardır. Bu grupta ortalama bekleme süresi yaklaşık 8.6 dk diğer kategorilere kıyasla daha düşük olmuştur. Ancak, bu vakaların sayıca fazla olması, doktor ve yatak kapasitesini etkileyebilecek yoğunluklar yaratabilir.

Orta dereceli vakalar, hizmet süresi ortalama 20 dakika olan ve toplam hastaların %30'unu oluşturan bir gruptur. Bu vakaların bekleme süresi (16.8 dk) hafif vakalara göre daha yüksektir. Çünkü daha uzun hizmet süresi ve kaynakların mevcut yoğunluğu, bu grupta hizmet verimliliğini azaltabilir. Orta vakalar için bekleme sürelerini azaltmak amacıyla doktor ve yatak kapasitesinin daha etkin tahsisi gerekebilir.

Ağır vakalar, toplam hastaların %20'sini oluşturmakta ve ortalama hizmet süresi 40 dakika olarak belirlenmiştir. Bu vakalar, yüksek öncelikli ve kritik bakım gerektiren durumlar olduğundan, diğer vakalara kıyasla daha uzun süre hizmet alırlar. Ağır vakalar için bekleme süresi 35 dk genellikle en yüksek değerleri alır, bu da kaynak tahsisinde dikkatli bir denge kurulmasını gerektirir. Bu tür vakalar için daha fazla doktor veya hemşire tahsisi yapılması, hizmet kalitesini artırabilir.

Sonuç olarak, simülasyon verileri, hastane kaynaklarının (doktor, hemşire ve yatak kapasitesi) her kategoriye uygun bir şekilde tahsis edilmesinin önemini göstermektedir. Hafif vakaların yüksek sayısı, orta vakaların daha uzun hizmet süresi ve ağır vakaların kritik önceliği göz önüne alındığında, kaynak planlamasında bu dağılım dikkate alınmalıdır. Bekleme sürelerinin düşürülmesi ve hizmet verimliliğinin artırılması için simülasyon modeli, kaynakların yeniden tahsisini optimize etmek için önemli veriler sunmaktadır.

MATLAB kodu çalıştırıldığında hasta bekleme sürelerinin histogramı aşağıdaki gibidir.



Grafik 7: Hasta Bekleme Sürelerinin Dağılımı

Burada mavi çubuklar hafif vakaların bekleme süresi, kırmızı çubuklar orta dereceli vakaların bekleme süresi ve yeşil çubuklar ağır vakaların bekleme süresini göstermektedir. Histogram grafiği, hafif, orta ve ağır vakaların bekleme sürelerini ayrı ayrı göstermektedir. Bu veriler, özellikle ağır vakaların bekleme süresinin daha uzun olduğunu açıkça ortaya koymaktadır. Bu durum, ek kaynakların (doktorlar, yataklar veya hemşireler) gerekliliğini gösterebilir.

Başlangıçta 3 doktor ile ortalama bekleme süresi 20,13 dakika çıktı, sonra doktor sayısını 4'e yükseldiğimizde ve kodu bir daha değiştirildiğimize görüyoruz ki optimizasyon sonrası 4 doktor ile ortalama bekleme süresi 18,3 dakikaya düştü. Ancak, ağır hastalar için bekleme süresi hala yüksek, bu yüzden hemşire sayısını da optimize etmek gerekebilir.

Bu simülasyonda, zaman aralığını genişleterek daha kapsamlı bir analiz yapmak mümkündür. Özellikle hastane operasyonlarının günün farklı dönemlerinde nasıl değiştiğini anlamak için günü üç bölüme ayırarak her biri için 8 saatlik simülasyon gerçekleştirebiliriz. Sabah, öğlen ve gece vardiyaları ayrı ayrı ele alındığında, hasta akışının yoğunluğu, bekleme süreleri ve kaynak kullanım verimliliği gibi faktörler daha detaylı analiz edilebilir. Böylece her vardiyaya özgü optimize edilmiş doktor, hemşire ve

yatak tahsis stratejileri belirlenebilir ve daha verimli bir hastane yönetimi sağlanabilir.

Günü üç vardiya olarak bölerek simülasyonu güncelleyelim ve her vardiya için analiz yapalım. Vardiyalar şu şekilde belirlenecektir.

- Sabah Vardiyası: 08:00- 16:00
- Öğlen Vardiyası: 16:00- 00:00
- Gece Vardiyası: 00:00- 08:00

Her vardiyada hasta geliş oranları farklı olabilir. Gün içinde yoğunluk sabah ve öğlen vardiyalarında daha yüksek olabilirken, gece vardiyasında daha düşük olması beklenir. Bu nedenle her vardiya için ayrı bir simülasyon çalıştırıp sonuçları karşılaştıracaktır.

MATLAB kodunu aşağıdaki girdi (input) değerler için tekrar çalıştırabiliriz.

Parametre	Sabah Vardiyası	Öğlen Vardiyası	Gece Vardiyası
Ortalama Hasta Geliş Hızı (hasta/saat)	18	15	8
Simülasyon Süresi (saat)	8	8	8
Hafif Vaka Hizmet Süresi (dk)	10	10	10
Orta Vaka Hizmet Süresi (dk)	20	20	20
Ağır Vaka Hizmet Süresi (dk)	40	40	40
Mevcut Doktor Sayısı	4	3	2
Mevcut Hemşire Sayısı	6	5	3
Mevcut Yatak Sayısı	12	10	8

Tablo 17: Girdi (Input) Değerleri

Buna göre çıktı (output) değerleri aşağıdaki tabelada verilmiştir;

Vardiya	Hasta Sayısı	Hafif Vaka Sayısı	Orta Vaka Sayısı	Ağır Vaka Sayısı	Ortalama Bekleme Süresi (dk)	Optimum Doktor Sayısı
Sabah	144	72	43	29	16.2	3.8≈ 4
Öğlen	120	60	36	24	17.8	4.2
Gece	64	32	19	13	14.5	2.5

Tablo 18: çıktı (Output) Değerleri

Sabah vardiyası, en yoğun hasta akışına sahip olduğundan dolayı doktor sayısını fazla tutmak gereklidir. Ancak simülasyon, mevcut doktor sayısının biraz fazla olduğunu gösteriyor, bu yüzden doktor sayısı 3'e düşürülebilir. Öğlen vardiyası, sabaha göre biraz daha az yoğun olmakla birlikte, bekleme süresi biraz daha yüksektir. Bu vardiya için doktor sayısının artırılması önerilir. Gece vardiyası, bekleme süresinin 15 dakikanın altında olduğu tek vardiyadır. Burada doktor sayısı az olsa bile hizmet yeterli seviyede sağlanmaktadır.

Simülasyon sonuçlarına göre şöyle bir önerilerde bulunabilir. Sabah vardiyasında doktor sayısını 4 yerine 3'e düşürmek kaynakların daha verimli kullanılmasını sağlayabilir. Öğlen vardiyasında doktor sayısını artırarak bekleme süresini 15 dakikanın altına çekmek mümkündür. Gece vardiyasında mevcut doktor sayısı yeterli görünüyor, bu yüzden ek kaynağa ihtiyaç yoktur. Vardiyalar arasında kaynak dağılımını dinamik olarak ayarlayacak bir sistem kurulabilir. Özellikle yoğun dönemlerde hasta sayısına bağlı olarak vardiya içi doktor rotasyonu düşünülebilir.

5.4 Yazılım Seçiminde Kriterler

Simülasyon yazılımı seçimi, modelleme yapılacak alanın gereksinimlerine, hesaplama kapasitesine, kullanım kolaylığına ve doğruluk seviyesine bağlı olarak dikkatli bir şekilde değerlendirilmelidir. Bu seçim sürecinde uygulama alanına bağlı olarak farklı kriterler öne çıkmaktadır. Örneğin, imalat sektörü için simülasyon yazılımı seçerken üretim süreçlerinin ayrık veya sürekli olup olmadığı, stok yönetimi, makine verimliliği ve üretim hatlarının optimizasyonu gibi unsurlar

dikkate alınmalıdır. Lojistik ve tedarik zinciri yönetiminde ise rota optimizasyonu, envanter yönetimi, taşıma süreleri ve dağıtım merkezlerinin konumlandırılması gibi faktörler önemlidir. Sağlık sektörü simülasyonlarında, hastaların bekleme süreleri, acil servis kapasitesi, personel planlaması gibi kriterler belirleyici olabilir.

Simülasyon yazılımı seçiminde teknik kriterler de büyük önem taşır. Matematiksel hesaplama gücü, özellikle büyük ölçekli sistemleri modellemek isteyen araştırmacılar için kritik bir faktördür. Örneğin, MATLAB ve AnyLogic, büyük ölçekli veri analizi ve optimizasyon için güçlü araçlar sunarken, Arena, ayrık olay simülasyonlarında daha esnek bir yapı sağlar. Simülasyonun görselleştirilmesi ve kullanıcı dostu arayüzü, özellikle endüstride kullanım için önem taşır. FlexSim ve Simul8, kullanıcı dostu arayüzleriyle ön plana çıkarken, detaylı programlama ve özelleştirme gerektiren projelerde MATLAB Simulink ve AnyLogic daha esnek çözümler sunabilir.

Ayrıca, yazılım seçimi yapılırken maliyet ve erişilebilirlik de göz önünde bulundurulmalıdır. Ticari yazılımlar, genellikle lisans maliyetleri gerektirirken, OpenModelica ve SimPy gibi açık kaynaklı alternatifler araştırmacılar için uygun çözümler sağlayabilir. Son olarak, simülasyon yazılımı seçiminde geçerlilik ve doğrulama süreçleri de göz önünde bulundurulmalıdır. Modelin doğruluğunu test edebilmek için, yazılımın hata analizleri, duyarlılık testleri ve senaryo analizleri yapmaya olanak tanıyıp tanımadığı dikkate alınmalıdır. Bu kriterlerin tamamı, simülasyonun başarılı bir şekilde gerçekleştirilmesi ve karar destek süreçlerinde etkin kullanılabilmesi açısından kritik bir rol oynamaktadır.

Yapılan araştırmalarda, kriterlere göre yazılım seçimi aşağıdaki tabelada özetleştirilmiştir.

Kriter	Öne Çıkan Yazılımlar	Kullanım Durumu
Teknik Kapasite	Arena, AnyLogic	Karmaşık süreçlerin simülasyonu, çoklu analiz modülü ve yüksek doğruluk gerektiren projeler.
Ekonomik Maliyet	Simio, ExtendSim	Düşük maliyetli projelerde veya kısa vadeli simülasyon ihtiyaçlarında kullanılır.
Kullanıcı Deneyimi	FlexSim, SIMUL8	Hızlı öğrenme ve sezgisel arayüz gerektiren projelerde veya yeni başlayan kullanıcılar için.
Özelleştirilebilirlik	AnyLogic, MATLAB	Yazılımda özel modeller oluşturulması ve esnek veri analizi gereken projeler.
Hızlı Çözüm Üretimi	SIMUL8, FlexSim	Kısa sürede sonuç alınması gereken durumlarda.
Güvenilirlik ve Uyum	Arena, AnyLogic	Kurumsal standartlara uyum sağlanması gereken projelerde veya farklı disiplinlerle entegrasyon.

Tablo 19: Yazılım Seçim Kriterleri

Bölüm 6. Gerçek Dünya Projeleri ve Simülasyon Uygulamaları

Simülasyon, akademik arařtırmalarda ve teorik analizlerde önemli bir yer tutsa da gerçek dünya projeleri üzerinde uygulanmadıkça etkisi sınırlı kalır. Simülasyon yöntemleri, farklı sektörlerdeki karmařık sistemlerin anlaşılmasını, optimize edilmesini ve karar destek süreçlerinin geliştirilmesini sağlar. Bu bölümde, saęlık, ulaşım, enerji ve lojistik gibi çeřitli sektörlerde gerçekteřtirilen gerçek dünya projeleri ele alınacak ve simülasyon tekniklerinin bu projelerde nasıl kullanıldığđı incelenecektir.

Gerçek projelerle çalıřmanın en büyük avantajđı, simülasyon modellerinin doęruluk ve güvenilirlięini artıran gerçek verilerin kullanılmasıdır. Açık veri kaynakları, kamu kurumları tarafından saęlanan istatistikler ve proje ortaklarından elde edilen özel veriler bu süreçte kritik bir rol oynar. Ancak, gerçek veri ile çalıřırken karřılařılan zorluklar da göz önünde bulundurulmalıdır. Veri güvenlięi, eksik veya hatalı veri problemleri ve sektörel dinamiklerin modellenmesi gibi konular, simülasyon projelerinin başarıya ulaşmasında belirleyici faktörlerdir.

Bu bölümde, bazı sektörler için özğün bir proje sečilerek, ilgili verilerin nereden temin edileceęi, nasıl işleneceęi ve hangi simülasyon yazılımlarının kullanılacaęı adım adım ele alınacaktır. Simülasyon sonuçlarının yorumlanması ve karar alma süreçlerine etkisi detaylandırılacak, ayrıca modelleme sürecinde karřılařılan zorluklar ve bunların nasıl ařılabileceęi tartıřılacaktır. Böylece, okurların simülasyonun pratik uygulamalarını daha iyi kavrayarak, kendi alanlarında benzer projeler geliřtirmeleri için saęlam bir temel oluřturulacaktır.

6.1 örnek 1: Türkiye’de Sel Tahminleri için Simülasyon Kullanımı

Bu projede, Türkiye’de meydana gelen ani sel baskınlarını tahmin etmek ve önlemek amacıyla bir simülasyon modeli oluřturacaęız.

Amacımız, belirli bölgelerde sel riskini analiz etmek ve erken uyarı sistemleri için öneriler geliştirmektir.

Sel felaketleri, özellikle yoğun yağış alan bölgelerde büyük can ve mal kayıplarına yol açmaktadır. Amacımız, geçmiş yağış verileri ve akarsu debi ölçümlerini kullanarak bir hidrodinamik simülasyon modeli kurmak ve kritik bölgeleri belirlemektir. Bu model, şehir planlamacıları ve yerel yönetimler için erken uyarı sistemleri geliştirmeye yardımcı olacaktır.

Bu proje kapsamında hedefler geçmiş yıllardaki yağış verilerini kullanarak sel tahminleri yapmak, şehir altyapısının sel baskınlarına nasıl tepki verdiğini simüle etmek ve hangi bölgelerde sel riski daha yüksek olduğunu belirlemektir.

Simülasyon modelimizin doğruluğunu artırmak için farklı veri setleri kullanmıştır.

Girdi (Input) Verileri:

Meteoroloji Verileri: Son 30 yıla ait saatlik ve günlük yağış miktarları, sıcaklık, nem ve rüzgâr verileri, bu veriler Türkiye Meteoroloji Genel Müdürlüğü'nden (MGM) veya açık veri kaynaklarından alınmıştır. <https://www.mgm.gov.tr>

Hidrografik ve Topografik Veriler: Nehir debi ölçümleri, toprak geçirgenlik oranları, yüzey eğimi ve arazi kullanımı bilgileri. Bu veriler Devlet Su İşleri (DSİ) ve Çevre, Şehircilik ve İklim Değişikliği Bakanlığı veri tabanlarından alınabilir.

Altyapı ve Şehir Planlama Verileri: Kanalizasyon ve drenaj sistemleri, yol, bina ve diğer altyapı bilgileri. Bu bilgiler yerel belediyelerden ve harita servislerinden sağlanabilir.

Bu projede kullanılabilecek üç temel simülasyon yöntemi vardır. Ajan Tabanlı Simülasyon (Agent-Based Modeling- ABM) , Hidrodinamik Modelleme (CFD - Computational Fluid Dynamics) ve Monte Carlo.

Bu projede, HEC-RAS ve MATLAB yazılımlarını kullanarak simülasyon yapacağız. MATLAB kullanarak hidrodinamik sel simülasyonunu basitleştirilmiş bir model ile gerçekleştirebiliriz. Aşağıda, MATLAB 'de sel tahmini yapmak için bir Monte Carlo simülasyonu ve

hidrodinamik hesaplama içeren bir kod sunmaktayız. Aşağıdaki kod, farklı yağış senaryolarında taşkın oluşumunu simüle eder ve risk seviyelerini belirler.

```
clc; clear; close all;

% Simülasyon Parametreleri
num_simulations = 1000; % Monte Carlo simülasyonu tekrar sayısı
rainfall_threshold = 50; % Taşkın eşiği (mm)
river_capacity = 200; % Nehir maksimum kapasitesi (m³/s)
drainage_capacity = 50; % Kanalizasyon kapasitesi (m³/s)

% Rastgele yağış üretimi (Monte Carlo Simülasyonu)
rainfall_data = normrnd(70, 20, [num_simulations, 1]); % Ortalama 70 mm yağış, 20 mm standart sapma

% Su seviyesi hesaplamaları
water_level = rainfall_data .* 0.03; % Yağışın yüzey akışına dönüşme katsayısı

% Taşkın olup olmadığını belirleme
flood_occurred = water_level > (river_capacity / 10);

% Taşkın olasılığı
flood_probability = sum(flood_occurred) / num_simulations;

% Sonuçları Görselleştirme
figure;
histogram(rainfall_data, 20, 'FaceColor', 'b');
xlabel('Yağış Miktarı (mm)');
ylabel('Frekans');
title(['Yağış Dağılımı ve Taşkın Olasılığı: ', num2str(flood_probability * 100), '%']);
grid on;

% Hidrodinamik Model (Basitleştirilmiş 1D Akış Modeli)
time = 0:1:24; % Saat cinsinden zaman
flow_rate = river_capacity * (1 + 0.2 * sin(2 * pi * time / 24)); % Debi değişimi

figure;
plot(time, flow_rate, 'r', 'LineWidth', 2);
xlabel('Zaman (Saat)');
ylabel('Debi (m³/s)');
title('Nehir Debisinin Günlük Değişimi');
grid on;

% Öneriler
if flood_probability > 0.3
    disp('UYARI: Taşkın riski yüksektir! Drenaj kapasitesi artırılmalıdır.ed(1:10), ...
else
    disp('Drenaj kapasitesi mevcut koşullar için yeterlidir.');
```

');

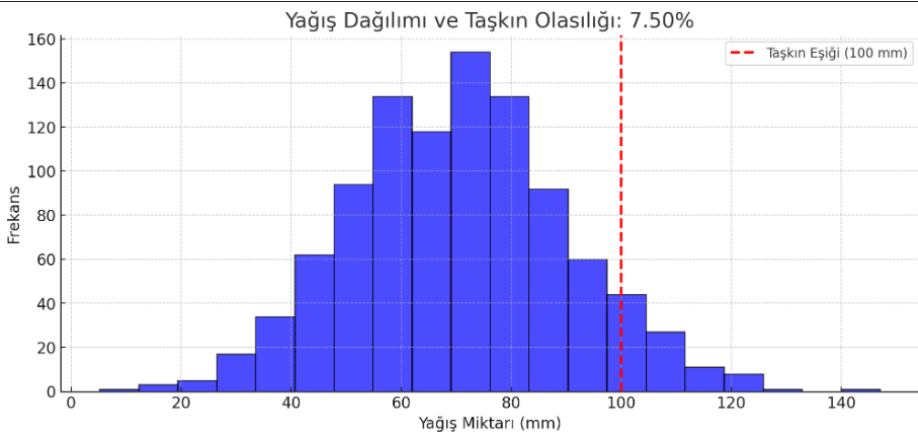
Grafik 8: Monte Carlo Simülasyon Kodu

Kod Açıklaması:

Monte Carlo Simülasyonu 1000 farklı yağış senaryosu oluşturulmuş, normal dağılıma sahip rastgele yağış miktarları simüle edilmiş ve su seviyesi hesaplanarak taşkın olup olmadığı belirlenmiştir. Hidrodinamik Modelinde ise günlük su debisinin zamana bağlı değişimi hesaplanıp nehir kapasitesine bağlı ($200m^3/s$) olarak taşkın risk analizi yapılır.

Kodda iki farklı grafik çiziliyor:

1. Yağış Dağılımı Histogramı:



Grafik 9: Yağış Dağılımı Histogramı

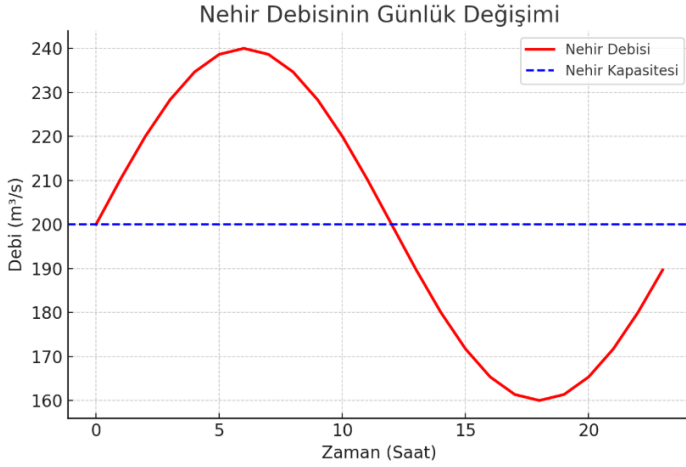
Bu grafik, Monte Carlo simülasyonu ile üretilen farklı yağış senaryolarının dağılımını gösterir. Burada X eksenı yağış miktarı (mm) ve Y eksenı frekanstır (kaç kez o yağış miktarı oluştu). Histogramın sağ tarafı (yüksek yağış değerleri), taşkın riskinin arttığı bölgeyi belirlemeyi yardımcı olur.

Histogramda ortalama yağış 70 mm civarında gözükür. 100 mm üzerindeki yağış değerleri az görülse de taşkın riskini artırıyor. Taşkın olasılığı %7.5 olarak hesaplandı, bu nedenle şehir planlamasında drenaj kapasitesini artırmak önemli.

2. Nehir Debisinin Günlük Değişimi:

Burada X ekseniz zaman (Saat) ve Y eksenı debidir (m^3/s). Dalgalı bir grafik (sinüs dalgası benzeri) çıkacaktır. Bu, gün içinde su akışının sabah ve akşam saatlerinde arttığını gösterir.

Debi sabit değil, gün içinde değişiyor. Sabah ve akşam saatlerinde ani artışlar var, bu da nehir akış hızının belirli saatlerde daha fazla olduğunu gösteriyor. Altyapı planlaması için kritik zaman dilimleri belirlenebilir (örneğin, sabah 8-10 ve akşam 18-20 saatleri arası).



Grafik 10: Nehir Debisinin Günlük Değişimi

Bu grafikler sayesinde hangi yağış miktarlarının risk oluşturduğunu, hangi saatlerde su akışının arttığını ve taşkın olasılığının ne olduğunu net bir şekilde görebiliriz.

6.2 örnek 2: Gazete Satıcısı Probleminin Simülasyonu

Gazete satıcısı probleminin simülasyonunda, modelin parametrelerini ARENA yazılımında en uygun şekilde seçerek, bir yıl boyunca satıcının ortalama kârını maksimuma çıkarmayı hedefliyoruz. Bu model, operasyon araştırmaları alanında yaygın olarak bilinen ve kullanılan modellerden biridir.

Gazete Satıcısı Problemi Nedir?

Bu simülasyon modelinde, günler üç kategoriye ayrılır: kötü gün, iyi gün ve orta gün. Her gün belirli bir olasılıkla bu kategorilerden birine girer.

İyi gün: Gazeteye yüksek talep olan günlerdir.

Orta gün: Ortalama talep seviyesine sahip günlerdir.

Kötü gün: Talebin düşük olduğu günlerdir.

Modelin temel amacı, her gün için gazete sipariş miktarını en uygun şekilde belirleyerek hem talebi karşılamak hem de gün sonunda satılmayan gazetelerin ziyan olmasını önlemektir. Bu sayede belirli bir zaman diliminde satıcının kârını maksimize etmeyi hedefliyoruz.

Modelin Parametreleri

Bu simülasyon modelinde kullanılan temel parametreler şunlardır: Gazetenin alış fiyatı, gazetenin satış fiyatı, farklı gün tiplerinin gerçekleşme olasılıkları, her gün tipi için gazete talebi miktarı.

Gazete satıcısı probleminin simülasyon modeli sayesinde, en iyi senaryoyu kolayca belirlemek mümkündür. ARENA yazılımında yer alan OptQuest aracı kullanılarak, en uygun senaryo belirlenebilir ve en yüksek kâr elde edilecek sipariş miktarı optimize edilebilir.

Ayrıca, bu tür problemler hem ayrık olay simülasyonu hem de ajan tabanlı modelleme ile çözülebilir. Dünyada bu tür modellemeler için farklı yazılımlar kullanılmaktadır. Örneğin, anyLogic, enterprise dynamics (ED), ARENA ve NetLogo.

Bu konu üzerine yapılan akademik çalışmalar arasında, "Analysis of Daily Newspaper Distribution in Solo by Agent-Based Simulation" başlıklı makalede, NetLogo yazılımı kullanılarak gazete satıcısı problemi ajan tabanlı modelleme ile çözülmüştür.

Bu modelde, gazete satıcısının ekonomik dengesini optimize etmeye çalışıyoruz. Modelde kullanılan temel parametreler şunlardır;

Gazetenin alış fiyatı	13\$
Gazetenin satış fiyatı	20\$
Satılmayan (iptal edilen) gazetenin maliyeti	2\$

Tablo 20: Gazete Satıcısının Simulasyondaki Parametreler

Üç farklı kategorideki günlerde belirli olasılıklara sahip olup, her günün gerçekleşme ihtimali farklıdır. Modelin amacı, her gün için en uygun sipariş miktarını belirleyerek satıcının en yüksek kârı elde etmesini sağlamaktır.

Gün Türü	Olasılık	Kümülatif Olasılık
İyi gün	0.35	0.35
Kötü Gün	0.45	0.8
Orta Gün	0.20	1

Tablo 21: Gün Türüne Göre Kar Olasılığı

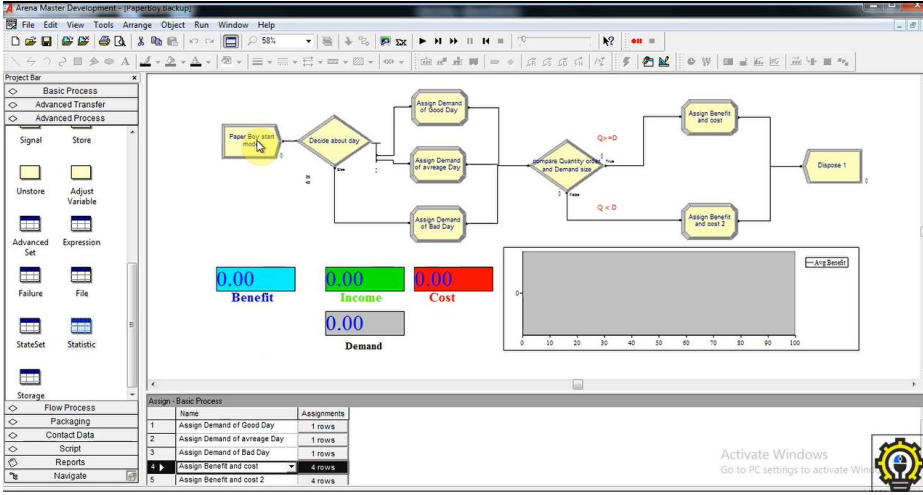
Her gün için talep miktarı farklıdır. Örneğin, iyi bir günde 40 adet gazetenin satılma olasılığı %3'tür. Kötü günlerde ise, örneğin 100 adet gazetenin satılma olasılığı sıfırdır, 90 adet gazetenin satılma olasılığı da sıfırdır. Bu şekilde, her gün türü için farklı talep seviyeleri aşağıdaki tabelada tanımlanmıştır.

Talep Miktarı	İyi Gün	Orta Gün	Kötü Gün
40	0.03	0.1	0.42
50	0.05	0.18	0.22
60	0.15	0.4	0.16
70	0.2	0.2	0.12
80	0.35	0.08	0.06
90	0.15	0.4	0
100	0.07	0	0

Tablo 22: Gün Türüne Göre Talep Miktarı Olasılığı

Amaç, kar miktarını maksimize etmektir. Bu yüzden, gazete satan çocuğun her gün ne kadar sipariş vermesi gerektiğini belirleyerek, bir yıl gibi belirli bir zaman aralığında ortalama kârını en yüksek seviyeye çıkarmayı hedefliyoruz.

Belirtilen koşullar doğrultusunda modeli oluşturup, çocuğun sipariş miktarını optimize etmeye çalışıyoruz.



Grafik 11: Gazete Satıcısının Arenada Simülasyon Modeli

Optimizasyonu başlatmak için “paper boy start modeli” in tıklayarak “Create Box” ile değerleri aşağıdaki gibi giriyoruz.

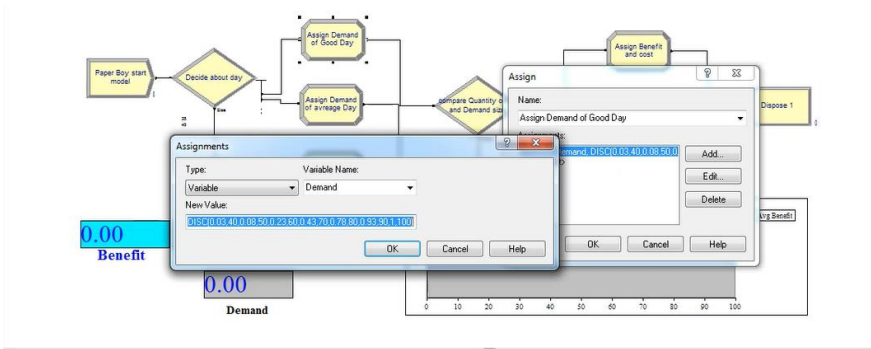
Grafik 12: Simülasyon Başlama değerlerin Girişi Create Box

Modelde “decide about day” kısmında her tip günün olasılıklarını giriyoruz.

“Assign demand of good day”, “Assign demand of bad bay” ve “Assign demand of average day” kutularda tabela 22’deki talep miktarını bir vektör olarak giriyoruz. Örneğin grafik 14 İyi günde talep miktarının veri girişlerini göstermektedir. Diğer günlerde benzer şekilde verilerini giriyoruz.

The screenshot shows the 'Decide' dialog box in NetLogo. The 'Name' field contains 'Decide about day' and the 'Type' is set to 'N-way by Chance'. The 'Percentages' list contains '35', '45', and '<End of list>'. The 'OK' button is highlighted with a yellow circle.

Grafik 13: Farklı Tip Günün Olasılıkları



Grafik 14: İyi Günde Talep Miktarı

Bir sonraki aşamada, maliyetleri ve gelirleri belirleyeceğiz. Bu maliyet ve gelirler, sipariş edilen miktarın günlük gazete talebiyle nasıl bir ilişki içinde olduğuna bağlıdır.

Örneğin, eğer sipariş miktarı talep miktarına eşit veya büyükse, hesaplamalar yukarıdaki senaryoya göre yapılır, aksi durumda ise farklı bir hesaplama yolu izlenir.

Sipariş miktarı talep miktarına eşit veya büyükse:

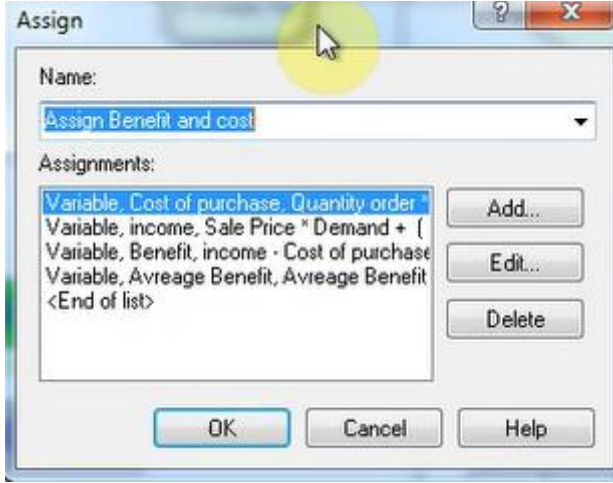
Satın alma maliyeti: Sipariş edilen miktar $\times$ Birim gazete maliyeti (13\$)

Satıştan elde edilen gelir: Satış fiyatı (Satış fiyatı 20\$)

× Talep edilen gazete sayısı

Fazla gazetelerin geliri: Fazla sipariş edilen ve satılmayan gazetelerin birim fiyatı 2 \$ olduğundan, bunların maliyeti hesaplanır. Sonuç olarak, tüm gelirlerden maliyetler çıkarılır ve kalan miktar kar olarak belirlenir. Bu kar negatif de olabilir.

Bütün bu değerleri “Assign benefit and cost” kısmına giriyoruz.

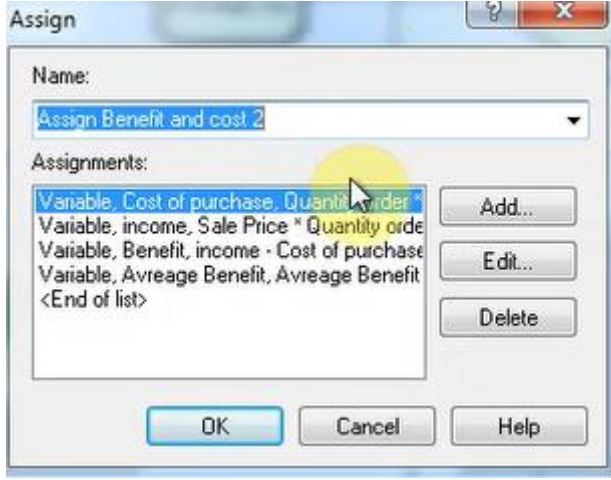


Grafik 15: Maliyetleri ve Gelirleri (Ek Gelir) İçin Veri Girişi

Sipariş miktarı talep miktarından küçükse:

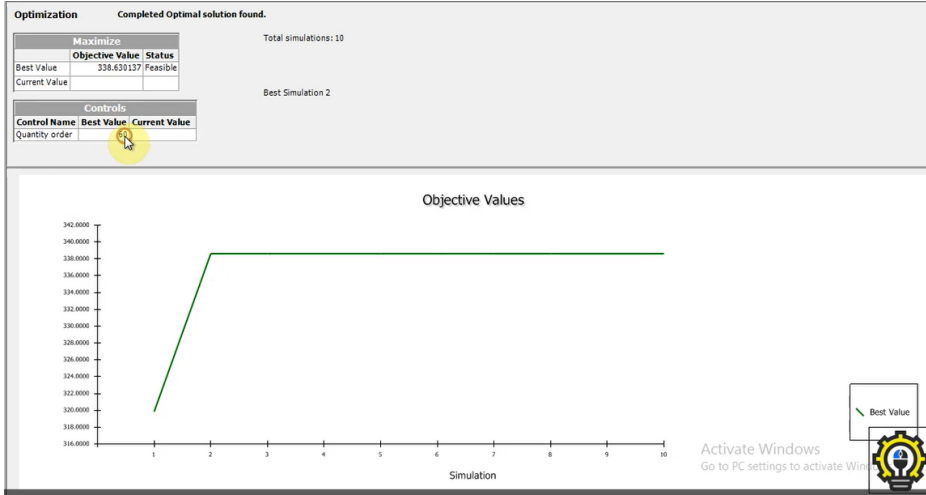
Yukarıdaki hesaplamalar tekrar edilir, ancak bu senaryoda satılmayan gazetelerden elde edilen ek gelir yoktur. Sonuç olarak, kar hesaplanır ve sistemden çıkış yapılır.

Bütün bu değerleri “Assign benefit and cost 2” kısmına giriyoruz.



Grafik 16: Grafik 15: Maliyetleri ve Gelirleri İçin Veri Girişi

Modeli çalıştırdığımızda, maliyetler atanır, talep belirlenir ve ortalama karın nasıl değiştiği grafik üzerinde gösterilir.



Grafik 17: Ortalama Karın Değişimi

Ayarları kârımızı maksimum yapmak ya da ortalama kârımızı maksimize etmek için belirliyoruz. Hedef de sipariş miktarını bulmaktır.

Modelin sonucuna göre, en uygun değer 60 oldu. Yani aslında, gazete satıcısı 60 gazete satmalı, böylece ortalama kârı maksimize olacak. Eğer 60 birim sipariş alırsa, ortalama kâr yaklaşık 338 \$ olacak. Bu da aslında farklı senaryoları çizdiği grafik, bunları da elde etmiş. Bu modelde, sipariş

miktarının sıfır ile 100 arasında olması gerektiğini belirledik. Yani minimum sipariş miktarımız sıfır, maksimum sipariş miktarımız ise 100. Siparişlerimiz de sadece 10'un katları olabilir. Bunu da kısıtlamalarımız arasında dikkate aldık. Bu yüzden sadece 10 senaryo buldu. Bu 10 taneyi inceledikten sonra, en iyi durum 60 oldu ve böylece hedefimiz olan ortalama kârı bir yıl içinde optimize etmiş oldu.

6.3 Örnek 3: Chicago O'Hare Bagaj Taşıma Sistemi

1992 yılında, Chicago'daki O'Hare Uluslararası Havalimanı'na yeni bir uluslararası terminal eklendi. Yeni terminal için bir bagaj taşıma sistemi gerekiyordu. Bu sistem hem bilet kontuarlarından gelen bagajları hem de aktarmalı uçuşlardan gelen bagajları taşıyacaktı. Bagaj taşıma konveyör sistemini tasarlamak için bir simülasyon kullanıldı. Modelin amacı, farklı konveyör düzenlerini ve bagaj depolama stratejilerini test etmektir.

Bazı uçuşlar için yolcular kalkıştan 3 saat önce havalimanına geldiğinden, bagajlarının bir kapı atanana kadar depolanması gerekiyordu.

Sistemdeki her bagaj, bir barkod etiketi ile tanımlanacaktı. Ancak bazı konveyör bölümleri saatte 360 feet (yaklaşık 110 metre) üzerinde bir hızla hareket ettiğinden, bagajları tanımlamak önemli bir sorun haline geldi. Eğer bir çanta tanımlanamazsa, bir operatör tarafından tekrar taranması için manuel kodlama alanına yönlendiriliyordu.

Sistemin işlediği yüksek bagaj hacmi nedeniyle, tarayıcıların küçük bir hata payı bile manuel kodlama alanında yığılmalara neden olabilir ve sistemin bir bölümünü devre dışı bırakabilirdi. Simülasyon, manuel kodlama alanında kaç operatörün gerekli olduğunu belirlemek için kullanıldı.

Chicago O'Hare Bagaj Taşıma Sistemi'nin simülasyonu, ayrık olay simülasyonu (Discrete Event Simulation- DES) yöntemiyle gerçekleştirilmiştir. Bunun nedeni, bagajların zaman içinde belirli olaylara (örneğin, konveyöre yüklenme, tarama, yönlendirme, manuel kodlama vb.) dayalı olarak hareket etmesidir. Operatör ihtiyacını tahmin etmek için Sistem Dinamikleri (System Dynamics- SD) kullanılmıştır.

Simülasyon süreci aşağıdaki gibidir;

Bagaj Akışı Modellenmesi:

- Yolcuların bagajlarını teslim etme süreleri
- Konveyör hızları (>360 ft/dak)
- Tarama süreçleri ve hata oranları

Manuel Kodlama Alanının Değerlendirilmesi:

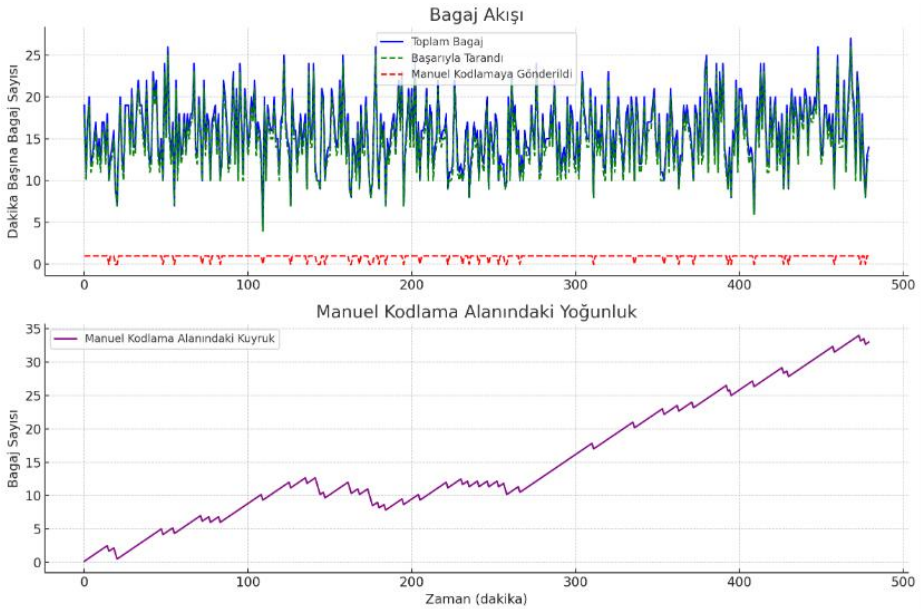
- Tarayıcıların no-read oranları (% kaç oranında barkod okunamıyor?)
- Manuel işlem için kaç operatör gerekiyor?

Farklı Konveyör Düzenlerinin Test Edilmesi:

- Konveyör darboğazları ve sıkışmalar analiz edildi.
- Bagajın yanlış yönlendirilme ihtimali değerlendirildi.

Python kullanarak bu bagaj taşıma sistemini simüle edilmiştir. SimPy (Simülasyon Kütüphanesi) kullanarak aşağıdaki sonuçlar elde edilmiştir.

Model, bagaj taşıma sistemindeki işleyişi benzeterek, kaç çanta işlendiğini, manuel kodlama alanındaki yoğunluğu ve sistemdeki darboğazları gösterecek.



Grafik 18: Bagaj Akışı ve Manuel Kodlama Alanındaki Yoğunluk

Sonuçlara göre bagaj akışı grafiğinde;

Mavi: Dakika başına toplam bagaj girişini gösterir.

Yeşil: Tarayıcıdan başarıyla geçen bagajlar.

Kırmızı: Manuel kodlamaya yönlendirilen bagajlar.

Ve manuel kodlama alanındaki kuyruk manuel işleme giden bagajların zamanla birikmesini gösteriyor ve ortalama 15 bagaj manuel kodlama kuyruğunda bekliyor.

Sonuç olarak manuel kodlama alanındaki iş yükü zamanla artıyor, kapasite yetersizse ek operatör ihtiyacı doğabilir ve daha iyi tarayıcılar kullanılarak manuel işleme giden bagaj sayısı azaltılabilir.

Özet olarak, 1992 yılında, Chicago O'Hare Uluslararası Havalimanı'na yeni bir uluslararası terminal eklendi ve bu terminal için bir bagaj taşıma sistemi tasarlandı. Bu sistemin tasarımında, farklı konveyör düzenleri ve bagaj depolama stratejilerini test etmek amacıyla bir simülasyon kullanıldı. Simülasyon, özellikle yüksek hızlı konveyör bölümlerinde (>360 ft/dak) bagajların barkodla tanımlanması ve tanımlanamayan bagajların manuel kodlama alanına yönlendirilmesi gibi kritik konuları ele aldı. Simülasyon sonuçları, manuel kodlama alanında kaç operatörün gerektiğini belirlemek ve sistemin genel verimliliğini artırmak için kullanıldı.

6.4 Örnek 4: Hastane Simülasyonu

Sağlık sektöründe alternatiflerin uygulanması özel bir zorluk teşkil etmektedir. Üretim sektöründe kararlar genellikle uygulama maliyeti ve yatırım getirisi etrafında şekillenir. Sağlık sektöründe ise maliyet önemli bir faktör olsa da her zaman en büyük sorun değildir. Simülasyon çalışmalarında genellikle ana hedef hasta işlem süresinin azaltılmasıdır. Sürenin bir maliyet faktörü vardır, ancak bu faktör genellikle belirsizdir. İşlem süresini azaltmak için ek ekipman veya personel gerekirse, zaman tasarrufunun gelir veya gider üzerindeki etkisini belirlemek zor (ve bazen tartışmalı) olabilir.

Bir çalışmaya karşı çıkanlar ekonomik temellere dayalı itirazlarda bulunabilirler. Bu, elbette, uygulamadan önce önerilen herhangi bir

çözümü sorgulamak için tamamen geçerli bir nedendir. Proje ekibi bu tür itirazları öngörmeli ve bunlara yanıt vermeye hazır olmalıdır. Simülasyon modeli, maliyeti etkileyen faktörlerin yakalanıp raporlanmasını sağlayacak şekilde tanımlanabilir. Bu sayede her alternatif için kademeli bir maliyet analizi yapılabilir.

PROMODEL Corporation'a ait MedModel simülasyon yazılımı kullanılarak bir acil servis bölümünde simülasyon çalışması gerçekleştirildi. Projenin başlangıcından tamamlanmasına kadar geçen süre 4 ay oldu. Model, bir hastane yeniden yapılandırma ekibinin projesinin bir parçası olarak tamamlandı.

Çalışmanın amaçları şunlardır:

1. Acil serviste hasta kalış süresini azaltabilecek eylemleri belirlemek.
2. Personel seviyelerini ve çalışma programlarını değerlendirmek.
3. Oda türlerine göre ihtiyaç duyulan oda sayısını belirlemek.
4. Dahili bir bekleme alanı eklenmesinin etkisini değerlendirmek.
5. Bir gözlem alanı eklenmesinin etkisini değerlendirmek.

Metodolojiye göre, sürecin bir sonraki adımı modeli tanımlamak ve modele girdi olarak kullanılacak verileri toplamaktır. Bu aşama yaklaşık 4 hafta sürdü. Acil serviste hastaların kalış süreleri hakkında bilgi edinmek için 1400'den fazla hasta gözlemlendi. Tablo 23 ile 25 veri toplama çalışmalarından elde edilen bazı derlemeleri göstermektedir (bu durumda hasta kategorileri, hastanın hastalık şiddetine göre sınıflandırılmıştır).

ZAM AN	PAZ AR	PAZART ESI	SA LI	CARSA MBA	PERSE MBE	CU MA	CUMAR TESİ	ORTALA MA
02:00	9.1	1.9	5.7	2.7	3.7	3.4	6.2	4.6
04:00	3.7	0.5	1.8	0.9	3.1	3.9	4	2.5
06:00	3.7	0.5	3.1	1.4	1.6	2.5	2.3	2.1
08:00	7	0	2.6	2.7	5.2	2.5	2.8	3.2
10:00	8	13.2	7.5	11.4	11	7.8	10.7	9.9
12:00	12.3	16	19. 8	12.7	16.2	12.3	18.1	15.4
14:00	13.4	15.1	8.8	12.3	13.1	12.3	7.9	11.8
16:00	11.8	7.1	7.5	13.6	11.5	16.2	8.5	10.9
18:00	8.6	16.5	14. 1	10.9	9.4	11.3	11.3	11.8
20:00	9.1	8	11	11.8	8.4	10.3	7.9	9.6
22:00	8	11.8	11	11.4	8.9	11.8	12.4	10.8
24:00	5.3	9.4	7	8.2	7.9	5.9	7.9	7.4

Tablo 23: Günü Farklı Zaman Dilimlerine Göre Hasta Geliş Yüzde (%)

Tablo 23, haftanın her günü için her 2 saatlik zaman diliminde gelen hastaların yüzdesini göstermektedir. MedModel'in hasta giriş düzenleyicisi kullanılarak, veri toplama sürecinde gözlemlenen günlük ve saatlik oranlara göre hastalar modele dahil edilmiştir.

Tablo 24, teşhis testi yapılan hastaların hastalık şiddetine göre yüzdesini özetlemektedir. Ventilasyon, yardımcı bir birim (bu durumda solunum terapisi) tarafından gerçekleştirilen bir işlemdir. Tablo 24'de bazı hastaların, ilk test sonuçları çıktıktan sonra ek laboratuvar testlerine ihtiyaç duyduğu unutulmamalıdır.

Benzer şekilde, bazı hastalar birkaç kez ilaç almaktadır ve bu durum Tablo 25'de gösterilmektedir. Tablo 25, bir hastanın acil servis ziyareti sırasında uygulanan prosedürleri (tedavileri) özetlemektedir. Bu prosedürler, acil servis personeli tarafından gerçekleştirilmektedir.

Seviye	Hasta Sayısı	X-Ray (%)	BT/US (%)	Laboratuvar (%)	Lab 1 Tekrar Test (%)	Lab 2 Tekrar Test (%)	Ventilasyon (%)	EKG (%)	Monitör (%)
Yetişkin	628	44.6	11	64.8	22.6	17.4	25	45.5	27.2
Acil	69	21.7	5.8	30.4	19	25	1.4	15.9	4.3
Subakut	271	45.4	10.7	62.7	22.9	15.4	0	30.3	5.9
Akut	280	48.9	12.9	74.6	22.5	17	5	66.1	51.4
Kritik	8	62.5	0	87.5	28.6	50	12.5	100	100
Pediyatri	477	9	1	15.9	13.2	20	0.8	2.1	1.7
Acil	255	2	0.4	7.1	11.1	50	0	0	0
Akut	214	17.3	1.9	24.8	11.3	16.7	0.9	3.7	2.8
Kritik	8	12.5	0	62.5	40	0	25	25	25
Hızlı Müdahale	336	8	1.2	12.5	9.5	25	0	3	0.3
Genel Toplam	1441	24.3	5.4	36.4	20.2	17.9	1.4	21.2	12.5

Tablo 24: Hasta Türüne ve Yardımcı Bölüme Göre Yardımcı Kullanımı

Seviye	Hasta Sayısı	IV (%)	İlaç 1 (%)	İlaç 2 (%)	İlaç 3 (% İlaç 2	Dikiş (%)	Bandaj (%)	NG Tüpü (%)	Foley Kateter (%)	Ortopedi (%)	Tromboliz (%)	Üçlü Tedavi (%)	Diğer (%)
Yetişkin	628	55.9	53	50.8	42.6	0.8	0.3	0.5	4	1.1	0.2	0.6	2.9
Acil	69	18.8	33.3	26.1	50	2.9	0	0	2.9	0	0	0	0
Subakut	271	51.3	44.3	49.2	32.2	0.7	0.4	1.1	3.7	2.2	0	0.4	2.6
Akut	280	68.6	66.1	55.1	48	0.4	0	0	4.3	0	0.4	1.1	3.6
Kritik	8	87.5	62.5	40	50	0	0	0	12.5	0	0	0	12.5
Pediyatri	477	8.8	26.6	28.3	30.6	2.3	0.6	0.4	0.8	0.8	0.4	0.2	2.1
Acil	255	2	19.6	8	0	0.4	0	0	0.4	0	0	0	0.8
Akut	214	15.9	33.2	39.4	32.1	4.7	1.4	0.5	0.9	1.9	0	0	3.3
Kritik	8	37.5	75	66.7	50	0	0	12.5	12.5	0	25	12.5	12.5
Hızlı Müdahale	336	3.3	15.5	55.6	40	1.8	0	0	0	1.1	0	0	2.1
Genel Toplam	1441	28	35.5	41.8	39.7	1.5	0.8	0.3	2	1	0.2	0.3	2.4

Tablo 25: Hasta Türüne ve Tedavi Türüne Göre Tedaviler (%)

Hastanenin tesisler departmanından temin edilen bir AutoCAD çizimi içe aktararak kat planı oluşturuldu. Grafik 19, bu kat planının bir kopyasını göstermektedir. Kat planı tamamlandıktan sonra, konumlar tanımlandı. Başlangıç modelinin oluşturulmasındaki bir sonraki adım, varlıkları, kaynakları, vardiya atamalarını ve hasta giriş desenlerini belirlemektir.

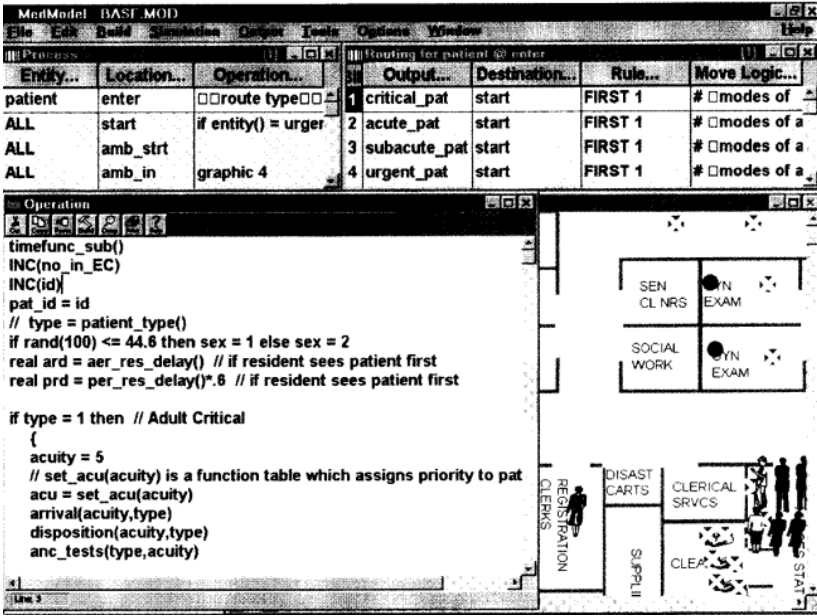
Hastalar ve diğer varlıklar arasındaki farklılıkları belirlemek için öznitelikler (attributes) tanımlandı. Bu öznitelikler, Tablo 24 ve Tablo 25’de belirtilen test veya tedavi olasılıklarına bağlı olarak 0 veya 1 (1 = doğru veya evet) değerleriyle atandı.

Başlangıç tanımlamaları tamamlandıktan sonra hastalar modele dahil edilebildi. Grafik 20, MedModel'in işlem düzenleyicisini göstermektedir. Bu modülde varlık akışı tanımlanır ve kontrol edilir. İşlem mantığı burada tanımlanır veya başlatılır (alt programlar veya makrolar çağrılabilir).

İşlem penceresi (Process Window), hangi varlığın işlendiğini ve işlemin nerede gerçekleştiğini tanımlar (örneğin, bir hastanın "Enter" adlı bir konuma gelmesi). Operasyon penceresi (Operation Window) ise hastanın Enter konumunda geçireceği süreçleri belirler. Bu örnekte, hastanın öznitelikleri atanır. Öznitelikler, ilerleyen aşamalardaki faaliyetler için varlık özelliklerini belirlemek amacıyla kullanılır.

Hasta özniteliklerine bazı örnekler:

- Hasta şiddet düzeyi (kritik hasta, acil hasta, acil olmayan hasta vb.)
- Hastanın bir asistan doktor tarafından muayene edilip edilmeyeceği
- Hastanın acil servise nasıl geldiği
- Hastanın eve mi gönderileceği yoksa hastaneye mi yatış yapılacağı
- Hastaya uygulanacak tedavi ve yardımcı testler



Grafik 20: İşlem Düzenleyicisi

(Yazar tarafından ProModel Corporation, Orem, Utah'a ait MedModel Yazılımı kullanılarak oluşturulmuştur.)

Yönlendirme penceresi, hastanın gönderileceği bir sonraki konumu tanımlar. Bu örnekte, hasta adlı genel bir varlık modele dahil edilmiştir ve olasılıklara göre bir hasta türü atanmıştır. Hasta daha sonra yeniden adlandırılır ve Start adlı bir konuma yönlendirilir.

Benzer şekilde modelin geri kalanı oluşturulmuştur. Analist modeli tamamladıktan sonra, ilk doğrulama süreci başlatılmıştır. Modele girilen geliş oranlarının hatasız olduğundan emin olunması gerekmiştir. Bu, model tarafından oluşturulan zaman serisi grafikleriyle kontrol edilmiştir. Bu grafikler gözlemlenen verilerle karşılaştırılmıştır. Günlük ve haftalık ortalama hacimler de doğrulanmıştır. Simülasyon modelinde uygulanan prosedürlerin ve testlerin sayıları, gözlemlenen prosedürler ve testlerle karşılaştırılmıştır. Gerçek hasta akışı, hasta türlerine göre adım adım orijinal akış diyagramlarıyla karşılaştırılmıştır. Karşılaştırmaya yardımcı olmak için izleme dosyaları ve hata ayıklama düzenleyicileri kullanılmıştır.

Analist, modelin doğru bir şekilde oluşturulduğundan emin olduktan sonra, aynı adımlar departman personeliyle birlikte de gerçekleştirilmiştir.

Analist doğrulamayı tek başına tamamlayamaz. İletişim hataları tespit edilemez, uygun olmayan varsayımlar da belirlenemezdi. Hatalar tespit edildikçe, analist hataları düzeltilmiş ve doğrulama işlemi hataların giderildiğinden emin olmak için tekrarlanmıştır.

Doğrulama tamamlandıktan sonra, geçerlilik süreci başlatıldı. Tablo 26, geçerlilik amacıyla toplanan gözlemlenen verilerin bir özetidir. Bu veriler aynı zamanda analiz amacıyla da kullanılmıştır. Dağıtım kararı, süreçte önemli bir noktadır. Hasta, attending physician (uzman doktor), hastayı yatırmaya ya da taburcu etmeye karar verene ve gerekli yönlendirmeleri ve talimatları belirleyene kadar tedavi odasından çıkamayacaktır.

Uzman ziyareti ile asistan ziyareti arasındaki süre, bir asistanın tanı kararı ve tedavi planı oluşturmak için gereken süredir. Asistan daha sonra bulgularını uzman doktora sunar, uzman doktor da asistan ile birlikte hastayı ziyaret eder.

Taburcu süresi, dağıtım kararının verildiği zamandan, hastanın gerçekten hastaneden ayrıldığı zamana kadar ölçülür. LOS (Length of Stay) ise hastanın hastanede kalış süresinin kısaltmasıdır.

Simülasyon modeli 8 gün boyunca çalıştırılmış ve 1 günlük ısınma (geçiş) süresi eklenmiştir. Otuz tekrar yapılmış ve sonuçlar raporlama düzenleyicisinde derlenmiştir. Gözlemlenen hastaların kalış süresi analiz edilerek buna bağlı güven aralıkları türetilmiş ve simülasyon sonuçları hipotez testi kullanılarak karşılaştırılmıştır. Başlangıçta, simülasyon modelinin sonuçları hastaların acil servisten çok hızlı ayrıldığını göstermiştir. Acil servis üzerindeki ek gözlemler ve ek görüşmeler, olası gecikmeleri belirlemek için gerekli olmuştur.

Orijinal akış diyagramları ile gözlemlenen varlık akışındaki farklar kaydedilmiş ve araştırılmıştır. Simülasyon modelinde değişiklikler yapılmış ve geçerlilik süreci tekrar edilmiştir. Model geçerli kabul edilmeden önce geçerlilik sürecinin iki ek tekrarı yapılması gerekmiştir (toplamda dört kez). Geçen süre yaklaşık 4 hafta olmuştur. Sürecin uzunluğunun bir kısmı, programların koordinasyonu ile ilgiliydi. Geçerliliği tamamlamak için yedi kişilik-gün gereklidir.

Alternatifler test edilip, sonuçlar birbirleriyle ve geçerlilik modeliyle karşılaştırılmıştır. Tablo 27, simülasyonla doğrulanan süreçte yapılan iyileştirmelerin kısa bir özetini sunmaktadır. Tablo 27'deki sonuçlar, uygulanmak üzere seçilen tüm önerilerin birleştirildiği bir modelden alınmıştır. Test edilen alternatifler şunları içermektedir (ama bunlarla sınırlı değildir):

- Hızlı müdahale alanındaki tedavi odalarının sayısını üç tane artırmak
- Hızlı müdahale alanına hasta yönlendirmek için triyaj kriterlerini gözden geçirmek (hızlı müdahale hacmini artırmak)
- Hastaların bakım takımlarına atama yöntemini revize etmek
- Acil servise bir gözlem alanı eklemek

Mod	Kayıt Zamanı	Trijaj Zamanı	Odaya Giriş ve Kadarsüre	Resident Ziyareti İçin Bekleme Süresi	Resident Ziyareti Süresi	Attending Ziyareti İçin Bekleme Süresi	Danışma Süresi	Dağıtım Kararı Süresi	Dağıtımdan Taburcuyla Kadar Süre	Toplam Kalış Süresi (LOS)
Tümü	32	12	63	82	86	220	150	129	257	257
Yetişkin Acil Servis										
Tümü	30	28	45	105	110	246	172	135	367	367
Yatış	28	45	33	112	101	280	188	198	497	497
Taburcu	32	19	54	103	110	213	155	29	284	284
Transferr	16	28	8	15	360	141	108	218	308	380
Pediyatrik Acil Servis										
Tümü	18	3	60	86	41	160	81	11	150	150
Yatış	13	8	27	147	3	240	114	178	340	340
Taburcu	18	2	63	81	52	95	54	47	134	134
Transferr	-	-	-	-	-	-	-	-	-	-
Hızlı Müdahale (Fast Track)										
Tümü	52	9	101	51	34	105	78	73	202	202
Yatış	69	8	126	15	469	491	305	827	827	827
Taburcu	52	9	100	51	34	70	38	12	182	182
Transferr	13	-	133	30	-	260	-	-	-	-

Tablo 26: Simülasyon Modelinin Geçerliliği İçin Derlenen Tarihsel Veriler

Ölçüt	Öncesi (dakika)	Sonrası (dakika)
Tüm hastalar için toplam kalış süresi (LOS)	257	181
Yetişkin acil servis kalış süresi (LOS)	367	329
Pediyatrik acil servis kalış süresi (LOS)	150	123
Hızlı müdahale (Fast Track) kalış süresi (LOS)	202	72
Ortalama hasta sayısı (20:00 itibarıyla)	36	28

Tablo 27: Simülasyon Sonuçlarının Özeti

Bu bölümde, farklı alanlardaki gerçek dünya problemlerine yönelik simülasyon uygulamalarını ele aldık. Her bir örnek, simülasyon tekniklerinin karar verme süreçlerini nasıl desteklediğini ve süreç iyileştirmelerinde nasıl kullanıldığını göstermektedir.

İlk olarak, Yağış Tahminleri için Simülasyon Kullanımı örneğinde, hava tahmini modellerinin doğruluğunu artırmak için simülasyon tekniklerinin nasıl kullanıldığını inceledik. Meteorolojik veriler ve istatistiksel modeller kullanılarak yapılan simülasyonlar, uzun vadeli yağış tahminlerinin daha güvenilir hale gelmesine yardımcı olmaktadır.

İkinci olarak, Gazete Satıcısı Probleminin Simülasyonu, stok yönetimi ve talep tahmini konusunda önemli bir örnek olarak karşımıza çıkmaktadır. Günlük talebin belirsiz olduğu bu senaryoda, simülasyon teknikleri kullanılarak en uygun sipariş miktarı belirlenmiş ve satıcının kârlılığını maksimize eden bir strateji geliştirilmiştir.

Üçüncü olarak, Chicago O'Hare Bagaj Taşıma Sistemi örneğinde, havalimanı bagaj taşıma süreçlerinin simülasyon yoluyla optimize edilmesi ele alınmıştır. Yüksek hızda hareket eden konveyör bantları ve manuel kodlama sistemleri için yapılan simülasyonlar, sistemin verimliliğini artırmak ve tıkanıklıkları önlemek amacıyla kullanılmıştır.

Son olarak, Hastane Simülasyonu ile sağlık sektöründe simülasyonun kullanımını değerlendirdik. Hasta akış yönetimi, acil servis operasyonları ve kaynak tahsis süreçlerinin simülasyon ile modellenmesi, hastanelerin

hizmet kalitesini artırmasına ve bekleme sürelerini minimize etmesine olanak tanımaktadır.

Bu dört örnek, simülasyonun farklı sektörlerde nasıl etkin bir karar destek aracı olarak kullanılabileceğini gözler önüne sermektedir. Gerçek dünya problemlerinin modellenmesi ve analiz edilmesi, sistem performansını iyileştirme noktasında büyük avantajlar sağlamaktadır. Simülasyonun sunduğu esneklik ve öngörü yeteneği, gelecekte de birçok alanda kritik rol oynamaya devam edecektir.

Bölüm 7. Simülasyon Projelerinde Başarı İçin Kılavuz

Dünya üzerindeki pek çok şey gibi, simülasyon projeleri de statik değildir; sürekli olarak gelişir ve kendilerini yeniden tanımlarlar. Bir proje ilerledikçe, yeni keşifler yapılır. Bazı keşifler projenin başlangıçtaki yönünü güçlendirirken, bazıları beklenmedik yön değişikliklerine yol açabilir. Bu değişimler kimi zaman küçük düzeltmeler olurken, bazı durumlarda projenin temel anlayışlarını bile kökten değiştirebilir. Ayrıca, proje sürecinde paydaşların ilgi alanları da değişebilir. Projeye başlarken önemli görülen konular zamanla geri planda kalabilir, yeni keşifler doğrultusunda farklı sorunlar öne çıkabilir.

Böylesine dinamik bir ortamda başarılı olmak, belirli ilkelere bağlı kalmayı gerektirir. Eğer bir simülasyon projesi sağlam bir yönetim çerçevesine oturtulmazsa, süreç yönsüz bir şekilde ilerleyebilir ve başarısızlık riski artar. Ancak, belirli yöntemler ve ilkeler doğrultusunda yönetilen projeler, hedeflerinden sapmadan daha istikrarlı bir şekilde ilerleyebilir. Ani ve keskin yön değişimleri azalır, sorunlar öngörülebilir bir çerçevede değerlendirilir ve proje daha sağlam bir zeminde yürütülür.

Bu bölümde ele alınan prensipler, başarılı bir simülasyon projesinin yönetilmesine yönelik rehber niteliği taşımaktadır. Süreç boyunca gerekli olan yönetsel becerileri öne çıkararak, proje yönetimi, model geliştirme ve analiz süreçlerini daha etkili bir şekilde yürütmeye yardımcı olur. Teknik detaylar doğrudan ele alınmasa da projenin başarılı olması için yönetsel süreçlerin en az teknik bilgi kadar önemli olduğu vurgulanmaktadır. Doğru yönlendirme ile teknik konular, daha az kesinti ve aksaklıkla sürdürülebilir. Sonuç olarak, proje performansı artar ve önerilerin uygulanma ihtimali yükselir.

7.1 Simülasyon Proje Adımları

Bir simülasyon projesi; analiz, geliştirme ve değerlendirme süreçlerinden oluşan bir dizi adımdan meydana gelir. Aşağıda listelenen adımlar, tüm simülasyon projeleri için temel bir çerçeve sunar ve sürecin düzenli bir akış içinde ilerlemesine yardımcı olur:

1. Problemin Tanımlanması – Projenin amacının ve ana hedeflerinin net bir şekilde belirlenmesi.
2. Veri Toplama ve Analiz – Simülasyon modelinin dayandırılacağı veri setlerinin seçimi ve incelenmesi.
3. Model Tasarımı ve Geliştirme – Simüle edilecek sistemin mantıksal ve matematiksel olarak modellenmesi.
4. Modelin Doğrulanması ve Geçerliliğinin Test Edilmesi – Modelin gerçek dünyayı ne kadar iyi temsil ettiğinin belirlenmesi.
5. Deneysel Senaryoların Çalıştırılması ve Sonuçların Analizi – Modelin çalıştırılması, farklı senaryoların test edilmesi ve sonuçların değerlendirilmesi.
6. Optimizasyon ve İyileştirme – Modelin en iyi sonuçları verecek şekilde optimize edilmesi.
7. Sonuçların Raporlanması ve Karar Destek Sürecine Entegrasyonu – Çıktıların ilgili paydaşlara sunulması ve uygulanabilir stratejilerin geliştirilmesi.

Her simülasyon projesi yukarıdaki sırayı tam olarak izlemeyebilir. Genellikle, proje ilerledikçe model tekrar gözden geçirilir ve süreç içerisinde bazı adımlar yeniden ele alınmak zorunda kalınır. Bu tür döngüsel geri dönüşler, simülasyon projelerinin doğasında vardır ve sürecin sağlıklı bir şekilde tamamlanması için gereklidir. Sonuç olarak, tüm adımların dikkatlice değerlendirilmesi, projeden elde edilecek çıktıların güvenilirliği açısından büyük önem taşır.

Problemin Tanımlanması

Bir projenin en kritik aşamalarından biri problemin doğru tanımlanmasıdır. Bu aşama, projenin diğer tüm faaliyetlerine rehberlik eder ve başarılı bir sonuca ulaşmak için temel oluşturur. Problemin tanımlanması aşağıdaki unsurları içermelidir:

1. Doğru başlangıç yapın.
2. Gerçek soruna odaklanın.
3. Beklentileri iyi yönetin.

4. Etkili sorular sorun.
5. Yargılamadan dinleyin.
6. Açık ve net iletişim kurun.
7. Çözümü öngörün.

1. Doğru Başlangıç Yapın

Proje ekibi, müşteriye ilk izlenimde güven vermelidir. Bu nedenle, projenin başlangıcında bir tanıtım toplantısı organize edilmesi faydalı olabilir. Bu toplantıda ekip tanıtılır, projenin amacı netleştirilir, sorumluluklar belirlenir ve iletişim yolları tesis edilir. İlk aşamada, teknolojiden ziyade sağlanacak faydalara odaklanmak daha etkili bir yaklaşımdır. Ayrıca, projenin başarıya ulaşması için ekibin bir liderinin (proje şampiyonu) bulunması gereklidir.

2. Gerçek Soruna Odaklanın

Yanlış tanımlanmış bir problem, projenin boşa zaman ve kaynak harcamasına yol açar. Bu nedenle, proje amacının net, ölçülebilir ve uygulanabilir olması gerekir. İlk etapta proje hedeflerini yazıya dökmek ve ekibin tüm çalışmalarını bu hedeflere göre yönlendirmesi gereklidir.

3. Beklentileri Yönetin

Müşterinin beklentileri, projenin doğru yönetilmesini sağlamak için baştan doğru şekilde belirlenmeli ve sürekli güncellenmelidir. Gerçekçi olmayan beklentiler, ekibin fazladan çaba harcamasına neden olabilir. Bu nedenle, modelin hangi sorulara cevap vereceği ve hangi sınırlamaların olduğu baştan açıkça belirtilmelidir.

4. Etkili Sorular Sorun

Etkili soru sormak, problemin kök nedenlerini anlamak için kritik bir beceridir. Sorular, müşterinin fark etmediği bilgileri açığa çıkarabilir veya yaratıcı çözümler üretmelerine yardımcı olabilir. Özellikle "Eğer sistemi değiştirme şansınız olsaydı, ne yapardınız?" gibi açık uçlu sorular, daha fazla bilgi edinmek için etkilidir.

5. Yargılamadan Dinleyin

Başarılı simülasyon uzmanları, iyi modelleme yeteneklerinden çok, iyi dinleme becerileriyle fark yaratırlar. Müşteriyi dikkatlice dinleyerek gerçek problemi anlamak önemlidir. İlk aşamada hızlı bir yargıda bulunmak, yanlış yönlere sapmaya neden olabilir. Müşterinin beklentilerini, önceliklerini ve endişelerini dikkate almak, başarıya giden yolda kritik bir fark yaratabilir.

6. Açık ve Net İletişim Kurun

Projelerin başarısız olmasının en yaygın nedeni, zayıf iletişimdir. Müşteri ile sürekli bilgi alışverişi yapılmalı, proje planı ve kritik aşamalar net bir şekilde aktarılmalıdır. Belirsizlikler ve sürprizlerden kaçınmak için, tüm gelişmeler zamanında bildirilmelidir.

7. Çözümü Öngörün

Projeye başlarken, problemi hızlı bir analizle incelemek faydalıdır. Bu, hem müşterinin soruna dair düşünmesini sağlar hem de ilerleyen aşamalarda karşılaşılabilecek farklılıkları anlamak için bir referans noktası sunar. İlk tahminler, projenin başlangıcındaki düşünce sürecini görmek ve ilerlemeyi ölçmek açısından çok önemlidir.

7.2 Model Kavramsallaştırma

Problem belirlenmesinin ardından, modelin tasarlanması aşamasına geçilir (bkz. Bölüm 2). Bu süreç, mevcut ya da planlanan sistemin önemli unsurlarının belirlenmesi ve bu unsurların bir simülasyon modeli olarak bütünleştirilmesini içerir. Bu sürecin başarılı olması için aşağıdaki ilkeler takip edilmelidir:

1. Modeli amaca uygun yönlendirin.
2. Hesaplanmış riskler alın.

Modeli Amaca Uygun Yönlendirme

Projenin hedefleri ve kapsamı belirlendikten sonra doğrudan model oluşturmaya başlamak cazip gelebilir. Ancak, bu aşama başlamadan önce modelin doğru çizgide ilerlemesini sağlamak kritik bir gerekliliktir. Modeli oluşturmadan önce şu sorulara yanıt bulmak gerekir: Model, sistemi birebir mi temsil etmeli yoksa daha soyut bir yaklaşımla mı ele

alınmalıdır? Bunun yanıtı, proje ekibinin elde etmek istediği sonuçlara bağlıdır.

Model, belirlenen amaca hizmet edecek şekilde tasarlanmalıdır. Örneğin, bir fabrikada iç birimlerin koordinasyonunu iyileştirmek amaçlanıyorsa, daha genelleştirilmiş bir model tercih edilebilir. Ancak, belirli bir hücrede sipariş sıralaması kritikse, ayrıntılı bir modelleme daha uygun olacaktır. Bu nedenle, modelleme kararları belirlenen hedeflere göre verilmelidir.

Modelin doğru yönlendirilmesi için en etkili yöntem "geri besleme" yaklaşımı kullanmaktır. Bu, öncelikle ana hedefleri belirleyip, bu hedefleri destekleyen kritik soruları listelemekle başlar. Örneğin, sistemin verimliliğini artırmak hedefleniyorsa, "Darboğaz oluşturan iş istasyonları hangileridir? Fazla kapasiteye sahip iş istasyonları var mı?" gibi sorular sorulmalıdır. Bu süreç, modelin odaklanması gereken unsurları belirleyerek daha verimli bir model oluşturulmasını sağlar.

Ayrıca, modelin sunduğu faydaların önceden belirlenmesi de büyük önem taşır. Çözümün sağlayacağı değeri önceden tanımlamak, elde edilen sonuçların etkisini artırır. Bunun yanında, modelin sınırlarını belirlemek ve gerekli girdi ölçülerini tanımlamak, modeli gereksiz ayrıntılardan arındırarak yalın ve etkili hale getirir.

Son olarak, modelin tüm bölümlerinin dengeli bir temsile sahip olması gerektiği unutulmamalıdır. Bir alanın fazla genelleştirilmesi, diğer alanlardaki hassasiyeti anlamsız hale getirebilir. Bu nedenle, verilerin ve mantıksal bağlantıların doğru bir düzeyde tutulması kritik bir faktördür.

Hesaplanmış Riskler Alın

Başarı, mevcut sınırları zorlamaktan geçer. Yeni şeyler denemek ve alışlagelmiş yaklaşımları sorgulamak, gelişimi destekler. Her projede farklı bir bakış açısı denemek, yeni bir simülasyon dili kullanmak veya farklı bir analiz yöntemi test etmek, gelişim için kritik adımlardır. Ancak, riskler faydalara oranla daha büyükse, geleneksel yöntemleri izlemek daha mantıklı olabilir.

Hesaplanmış riskler almak, bazen başarısızlıklarla karşılaşmayı da beraberinde getirir. En iyi projeler bile engellerle karşılaşabilir. Burada

önemli olan, ekibin bu başarısızlıklardan ders çıkararak ilerleyebilmesidir. Başarısızlıklar, yeni fırsatların kapısını açabilir ve ekibi daha yenilikçi çözümler bulmaya yönlendirebilir.

Peki, başarısızlıklar ne zaman raporlanmalıdır? Müşteriyi hemen bilgilendirmek mi yoksa önce bir çözüm bulup mu paylaşmak gerekir? Bu durum proje özelinde değişebilir. Ancak, genel kural olarak, hiç kimse bir sorunu en son öğrenmek istemez. Müşterileri zamanında bilgilendirmek, güven oluşturur ve projenin daha sağlam bir temel üzerinde ilerlemesini sağlar. Erken uyarı sistemleri her zaman kusursuz olmayabilir, ancak herhangi bir sistemin hiç olmamasından daha iyidir.

7.3 Veri Toplama

Tüm simülasyon modelleri veri gerektirir (bkz. Bölüm 3). Sistem hakkında bilgi toplamak ya da sistem henüz mevcut değilse tahminlerde bulunmak zorunludur. Ayrıca, sistem daha iyi anlaşıldıkça ek veriler talep edilebilir, bu nedenle veri toplama işlemi sürekli tekrarlanabilir.

Veri toplama sürecinde dikkat edilmesi gereken hususlar:

- Verileri sorgulayın.
- Varsayımlar yapın.

Verileri Sorgulamak

Toplanan tüm verileri eleştirisel bir bakış açısıyla değerlendirin. Verinin kaynağı, hangi koşullarda ve ne zaman toplandığı, hangi yöntemlerin kullanıldığı gibi faktörleri dikkate alarak bir denetim yapın. Verinin mantıklı olup olmadığını ve proje kapsamına uygunluğunu değerlendirin.

Eksik, hatalı veya tutarsızlık gösteren veriler, çalışmanın sonuçlarına güveni zedeleyebilir ve bu da çalışmanın etkisini azaltabilir. Bu nedenle, kullanılan verilerin modelleme yapılan süreci gerçekçi bir şekilde temsil ettiğinden emin olun.

Verileri eksiksiz toplamak önemlidir ancak gereksiz ayrıntılara takılıp proje sürecini aksatmamak gerekir. En iyi yaklaşım, mevcut verilerle başlamak ve ihtiyaç duyuldukça ek bilgiler toplamaktır. Bu şekilde veri toplama ve modelleme paralel ilerleyebilir.

Varsayımlar Yapmak

Veri toplama sürecinde eksik bilgiler olabilir. Bu durum projenin ilerlemesini engellememelidir. Bilinmeyen noktalar için mantıklı varsayımlar yaparak modeli oluşturmaya devam edin. Varsayımlarla modellenen verinin sisteme etkisini değerlendirerek doğruluğunu sıklıkla kontrol edin.

Bir varsayım başlangıçta yapıldı diye sabit tutulmak zorunda değildir. Proje ilerledikçe yeni bilgiler elde edildikçe varsayımların güncellenmesi gerekir. Sonuçları sunmadan önce tüm varsayımları tekrar gözden geçirin ve gerçek verilere dayanarak doğrulamalar yapın.

7.4 Model Oluşturma

Simülasyon modeli, daha önceden oluşturulan konsept taslağı temel alınarak inşa edilir. Ancak aynı taslak üzerinden çok farklı modeller geliştirilebilir. Modelleme sürecinde dikkat edilmesi gereken hususlar:

- Probleme odaklanın.
- Basit başlayın.
- Karmaşıklığı kontrol altında tutun.
- Proje ilerleyişini sürekli sürdürün.
- Sürekli gözden geçirin.
- Probleme Odaklanmak

Birçok simülasyon projesi, model oluşturmaya gereğinden fazla odaklanır ve esas problem unutulabilir. Modelin amaçlarına uygun çözümler bulmak en önemli hedeftir. Modeli oluşturduktan sonra, farklı senaryoları test etmek için yeterli zaman bırakın.

Basit Başlamak

Karmaşık modellerin çözüm bulmayı zorlaştırdığı unutulmamalıdır. Gereksiz ayrıntılarla başlamak, modelin verifikasyonunu ve anlaşılabilirliğini zorlaştırabilir. Bunun yerine, sistemin temel yapısını yansıtan basit bir model oluşturun ve zamanla detay ekleyerek genişletin.

Karmaşıklığı Kontrol Altında Tutmak

Simülasyon modelinin çok karmaşık hale gelmesi modelin gerçek sistemi gizlemesine neden olabilir. Model, gerçek sistemin eksikliklerini

gizlemek yerine bu eksiklikleri ortaya çıkarmalıdır. Modelleme yapılırken sistemin gerçek koşullarına uygun olması önemlidir.

Proje İlerlemesini Sürdürmek

Bir simülasyon projesinin ilerlemesini görünür kılmak çok önemlidir. Proje süresince düzenli olarak ilerleme raporları ve ara sunumlar yapmak, proje ekibinin ve paydaşların motivasyonunu yüksek tutar.

Sürekli Gözden Geçirmek

Modelleme sürecinde belli aralıklarla durup projeyi değerlendirmek önemlidir. Ekibin çalışmasının amaca uygun ilerleyip ilerlemediğini sorgulayın. Gereksiz çalışmalar yapılıyorsa bunları yeniden planlayın. Aylık veya belirli kilometre taşlarında proje değerlendirme toplantıları yaparak ilerlemeyi izleyin.

Son söz olarak, simülasyon projelerinde başarılı olmak için veri toplama, modelleme ve doğrulama süreçlerinin birbirini desteklemesi gerekir. Modeli gerçek problemlere odaklı tutarak, karmaşıklığı kontrol ederek ve düzenli olarak güncelleyerek en iyi sonuçlar elde edilebilir.

7.5 Doğrulama ve Geçerleme

Doğrulama aşamasında, modelin gerçekleştirilen işlemleri test edilir (bkz. Bölüm 10). Bu, verilerin doğru şekilde okunduğunun ve varlıkların öngörülen şekilde akışının sağlandığını kontrol etmeyi içerir. Modelin doğrulandığından emin olunduktan sonra geçerleme aşamasına geçilir. Burada, model gerçek performans ile simüle edilen performansın makul bir uyum göstermesi için kalibre edilir veya önerilen sistemler söz konusuysa beklenen performans ile karşılaştırılır. Bu aşamada dikkate alınması gereken temel ilkeler şunlardır:

1. Değişiklikleri kontrol altında tutun.
2. Müşteri algılarını göz önünde bulundurun.

Değişiklikleri Kontrol Altında Tutun

Projeler ilerledikçe, yeni edinilen bilgiler modelin tekrar gözden geçirilmesini gerektirir. Bu durum, müşterilerin değişiklik taleplerinde bulunmalarına yol açabilir. Bu nedenle, bu taleplerin gelecekte karşınıza

çıkacağını öngörün ve bunları etkin bir şekilde yönetmek için bir plan hazırlayın.

Proje başlangıcında küçük gibi görünse de her değişiklik isteğine olumlu yanıt vermek uzun vadede büyük sorunlara neden olabilir. Proje ekibi, projenin amacı ve kapsamı ile uyumlu olmayan değişikliklere karşı şartlar uygun olduğunda itiraz etmelidir. Çünkü basit görünse de, bir değişikliğin gerçek etkileri genellikle ancak projenin ilerleyen aşamalarında ortaya çıkar ve büyük gecikmelere yol açabilir.

Değişiklikleri ertelemek bazen iyi bir strateji olabilir. Mevcut plana göre ilerleme sağlandıktan sonra değişiklikler gözden geçirilebilir. Ayrıca, müşterinin tüm değişiklik toplantılarına dahil edilmesi hem projeye olan bağlılığını artıracak hem de yapılan değişikliklerin etkilerini daha iyi anlamasını sağlayacaktır.

Müşteri Algılarını Göz Önünde Bulundurun

Bir proje ekibi, önceden kazandığı tecrübelerle belirli bir işleyiş düzeni geliştirir. Ancak müşteriler farklı bir bakış açısına sahip olabilirler. Ekip için önemsiz gibi görünse de, müşteriler için bazı hatalar büyük bir endişe kaynağı olabilir. Örneğin, göz ardı edilen küçük bir programlama hatası, müşterinin modele duyduğu güveni zedeleyebilir ve geçirme sürecini gereksiz yere uzatabilir. Bu nedenle, müşteri tarafından dile getirilen hatalar hızlıca düzeltilmelidir.

7.6 Analiz

Simülasyon modelinde yapılan deneyler, belirlenen hedefler doğrultusunda çıkarımlar yapılmasını sağlar. Bu süreçte farklı analiz yöntemleri kullanılabilir (bkz. Bölümler 6-9). Analiz yapılırken dikkat edilmesi gereken temel ilkeler şunlardır:

1. Modeli aktif olarak kullanın.
2. Çıktıları sorgulayın.
3. Modelin sınırlarını bilin.
4. Ne zaman duracağınızı bilin.
5. Alternatifler sunun.

6. Başarıyı vurgulayın.

Modeli Aktif Olarak Kullanın

Simülasyon modeli, sistemin işleyişini farklı bakış açılarından incelemeye olanak tanır. Model sadece bir araçtır; asıl değeri, proje ekibinin onu ne kadar etkili kullandığına bağlıdır. Farklı senaryolar üzerinde çalışarak sistemin davranışına dair öngörüler elde edilebilir.

Çıktıları Sorgulayın

Modelin çıktılarının mantıklı olup olmadığını kontrol edin. Sonuçlar beklenen değerlerle uyumlu mu? Aksi durumda, modelin varsayımlarını ve geçerleme sürecini tekrar gözden geçirin.

Alternatifler Sunun

Tek bir çözüm sunmak, karar alma sürecini sınırlandırabilir. Bunun yerine, birden fazla alternatif sunarak müşterinin sürece daha fazla dahil olmasını sağlayın.

Başarıyı Vurgulayın

Projeyi başarılı bir şekilde tamamlamak kadar, bu başarıyı etkili bir şekilde sunmak da önemlidir. Müşteriye, projenin sağladığı değeri sürekli hatırlatın ve elde edilen kazançları vurgulayarak projenin olumlu yönlerini öne çıkarın.

7.7 Dokümantasyon

Simülasyon modelleri kullanım için oluşturulur ve zamanla gelişir. Modelin ve projenin eksiksiz bir şekilde dokümente edilmesi, modelin uzun vadede kullanılmasını desteklemek ve gerektiğinde modelleme ile analiz çalışmalarını genişletmek için gerekli bilgileri sağlar. Bu süreçte dikkate alınması gereken temel ilkeler şunlardır:

1. Standartlara bağlı kalın.
2. İlerlemeyi raporlayın.

Standartlara Bağlı Kalın

İyi programlama ve dokümantasyon standartlarına uymak her zaman önemlidir. Bazı simülasyon modelleri yalnızca belirli bir sorunu incelemek

için geliştirilir ve bir daha kullanılmazken, bazı modeller ise mevcut ve gelecekteki pek çok sorunu ele almak için oluşturulur. Uzun ömürlü modeller, onlarca yıl boyunca geçerliliğini koruyabilir. Bu tür modellerin düzenli ve sistematik bir şekilde dokümanite edilmesi, modelin sürdürülebilirliğini sağlar.

Öte yandan, kısa ömürlü modeller için bu gereklilik göz ardı edilebilir gibi görünse de, modelin gelecekte tekrar gündeme gelmeyeceği garanti edilemez. Geçmişte "hızlı ve geçici" olarak düşünülen birçok modelin yıllar sonra yeniden ele alındığı durumlar olmuştur. Eğer yeterli dokümantasyon sağlanmamışsa, modeli doğrulamak ve geliştirmek büyük bir çaba gerektirebilir. Bu nedenle, modelin ne kadar süre kullanılacağı öngörülemezse bile, iyi programlama ve dokümantasyon uygulamalarına her zaman sadık kalınmalıdır.

Standartlara bağlı kalmak, ekip üyeleri arasındaki iletişimi güçlendirir, hata ayıklama süresini azaltır ve modelin bakımını kolaylaştırır. Genellikle müşteri tarafından doğrudan fark edilmese de, bu süreç uzun vadede projenin başarısını büyük ölçüde etkiler.

İlerlemeyi Raporlayın

Proje sürecinin yazılı olarak kayıt altına alınması, yapılan çalışmaların ve alınan kararların kronolojik bir kaydını oluşturur. Bu tür kayıtlar, projenin doğru yönde ilerlemesini sağlamak açısından çok değerlidir.

İlerleme raporları en az ayda bir kez hazırlanmalıdır. Bu sayede, projeye doğrudan dahil olmayan kişiler bile süreç hakkında bilgi sahibi olabilir. Düzenli raporlama, yanlış anlamaların erken tespit edilmesini ve olası problemlerin düşük maliyetle çözülmesini sağlar.

Ayrıca, bir proje günlüğü tutmak da büyük önem taşır. Bu günlükte; tamamlanan işler, karşılaşılan önemli sorunlar, değişiklik talepleri, kritik kararlar ve gelecekte yapılması planlanan çalışmalar detaylı bir şekilde kaydedilmelidir. Özellikle proje sürecinde hedefler, kapsam, zaman çizelgesi veya ekip üyeleri değişirse, bu tür kayıtlar büyük bir rehber niteliği taşır.

7.8 Uygulama

Bir simülasyon projesinin nihai amacı, müşterinin somut bir eylem gerçekleştirmesini sağlamaktır. Eğer yapılan çalışmalar sonucunda sistemin performansı iyileşirse, proje başarılı kabul edilir. Bu aşamaya ulaşabilmek için aşağıdaki temel prensipler dikkate alınmalıdır:

1. Güven oluşturun.
2. Kazanmaya odaklanın.
3. Takım çalışmasını teşvik edin.
4. Kritik paydaşları sürece dahil edin.
5. Sunumları etkili bir şekilde yapılandırın.
6. Sürekli iyileştirme anlayışını benimseyin.
7. Projeyi tamamladıktan sonra takip sağlayın.

Güven Oluşturun

Müşteri ile ekip arasındaki en önemli unsur güvendir. Güven, sözle değil, eylemlerle kazanılır. Ekip, yüksek etik standartlara bağlı kalmalı ve taahhüt ettiği her şeyi eksiksiz yerine getirmelidir. Güven kazanmak zor, kaybetmek ise çok kolaydır. Bu yüzden dikkatle korunmalıdır.

Kazanmaya Odaklanın

Projelerin başarısız olma nedenleri genellikle yanlış anlamalar, yetersiz çaba veya eksik uygulamalardır. Başarıya ulaşmak için ekibin olumlu bir bakış açısına sahip olması gerekir. Olumlu düşünmek, başarı beklentisini artırır ve ekibi motive eder.

Takım Çalışmasını Teşvik Edin

Simülasyon projeleri yalnızca model geliştirmekten ibaret değildir; insanlarla etkili bir şekilde çalışmayı da gerektirir. Ekip üyeleri arasındaki güven ve saygı, iş birliğini güçlendirir. Ekip içindeki çatışmalar en aza indirilmeli ve herkesin proje sürecine katkı sağlaması teşvik edilmelidir.

Kritik Paydaşları Sürece Dahil Edin

Projeye etki edebilecek karar alıcıları ve süreç sahiplerini en baştan sürece dahil etmek, sonradan yaşanabilecek olası engelleri ortadan kaldırır. Paydaşların projeye olan ilgisi ve desteği sağlanmalı, belirli aralıklarla bilgilendirme yapılmalıdır.

Sunumları Etkili Yapılandırın

Projenin çıktıları ne kadar güçlü olursa olsun, sunum tekniği zayıfsa mesaj doğru iletilmeyebilir. Sunumlar kısa, net ve etkili olmalıdır. Görsel öğeler kullanılmalı ve gereksiz detaylardan kaçınılmalıdır.

Sürekli İyileştirme Anlayışını Benimseyin

Simülasyon projeleri genellikle sistem iyileştirmeleriyle sonuçlanır. Ancak değişikliklere karşı dirençle karşılaşmak olasıdır. Bu nedenle, yönetimin desteği alınmalı, ekip değişimin gerekliliği konusunda eğitilmeli ve herkesin sürece katılımı teşvik edilmelidir.

Projeyi Tamamladıktan Sonra Takip Sağlayın

Proje sona erdiğinde müşteriyle bağlantıyı kesmemek önemlidir. Proje çıktılarının uygulanıp uygulanmadığını, performans değişikliklerini ve yeni sorunların ortaya çıkıp çıkmadığını takip etmek, müşteriyle uzun vadeli bir iş ilişkisi geliştirmek açısından faydalıdır.

Her projede yaşanan deneyimler farklıdır. Bir simülasyon uzmanı için doğru olan bir yöntem, bir başkası için geçerli olmayabilir. Ancak, zamanla bazı temel ilkelerin değişmediği görülür. Projelerin özellikleri farklılık gösterse de, başarılı projelerin ortak noktası belirli prensiplere bağlı kalmalarıdır.

Bu ilkeler, projenin daha planlı ve odaklı ilerlemesine yardımcı olur ve başarı olasılığını artırır. Sonuç olarak, simülasyon projelerinde en önemli gerçek, müşteri değerini ön planda tutan bir anlayışa sahip olmaktır. Model doğruluğu, analiz kapsamı veya zaman çizelgesi gibi unsurlar önemlidir, ancak uzun vadede en değerli olan şey, müşterinin memnuniyetidir.

Bölüm 8. Gelecek Perspektifleri

Simülasyon teknolojileri, bilimsel araştırmalardan endüstriyel uygulamalara kadar geniş bir yelpazede giderek daha kritik bir rol oynamaktadır. Günümüzde yapay zekâ, büyük veri, dijital ikizler ve kuantum bilişim gibi alanlardaki ilerlemeler, simülasyon modellerinin hem doğruluğunu hem de etkinliğini artırmaktadır. Bu gelişmeler, simülasyonun sadece bir analiz aracı olmaktan çıkıp, karar verme süreçlerinin ayrılmaz bir parçası hâline gelmesini sağlamaktadır. Bu bölümde, simülasyonda yapay zekâ ve makine öğrenmesi, dijital ikizler ve Endüstri 4.0, multidisipliner simülasyonun geleceği, kuantum bilişim ve simülasyon, etik ve sürdürülebilirlik gibi konular ele alınacaktır. Yeni nesil simülasyon yaklaşımlarının sunduğu fırsatlar ve karşılaşılabileceğimiz zorluklar tartışılarak, bu alandaki potansiyel gelişmeler hakkında bir bakış açısı sunulacaktır.

8.1 Simülasyonda Yapay Zekâ ve Makine Öğrenmesi

Simülasyon, karmaşık sistemlerin modellenmesi, analiz edilmesi ve optimize edilmesi için uzun yıllardır kullanılan temel bir yöntemdir. Günümüzde, yapay zekâ (YZ) ve makine öğrenmesi (MÖ) tekniklerinin gelişimi, simülasyon süreçlerine yeni bir boyut kazandırmış ve daha verimli, ölçeklenebilir, esnek modellerin oluşturulmasına olanak sağlamıştır. Bu entegrasyon, geleneksel simülasyon yöntemlerinin sınırlarını genişleterek, sistemlerin daha hızlı ve doğru bir şekilde analiz edilmesine olanak tanımaktadır.

8.1.1 Yapay Zekâ ve Makine Öğrenmesinin Simülasyona Katkıları

YZ ve MÖ, simülasyon süreçlerine çeşitli açılardan katkı sağlamaktadır:

1. Model Karmaşıklığının Yönetimi: Geleneksel simülasyon modelleri, büyük veri kümeleri ve karmaşık sistemler karşısında hesaplama açısından maliyetli olabilmektedir. MÖ algoritmaları, büyük veri kümelerinden örüntüler çıkararak simülasyon süreçlerini daha verimli hale getirmektedir.

2. Simülasyonun Hızlandırılması: Derin öğrenme tabanlı yöntemler ve azalan hesaplama maliyetleri sayesinde simülasyon süreçleri daha hızlı

hale getirilebilir. Özellikle fizik tabanlı simülasyonlarda, veri odaklı yaklaşımlar kullanılarak geleneksel hesaplamalara göre daha kısa sürede sonuç elde edilebilir.

3. Öngörücü Simülasyon ve Optimizasyon: Makine öğrenimi, simülasyon süreçlerine öngörücü analitik yetenekler ekleyerek olası senaryoların incelenmesini ve sistemlerin optimize edilmesini mümkün kılar. Genetik algoritmalar, pekiştirmeli öğrenme ve derin öğrenme gibi teknikler, karar verme süreçlerini destekleyerek sistem performansını artırabilir.

4. Gerçek Zamanlı Adaptasyon: Simülasyonlar, dinamik sistemlerin modellenmesinde sıkça kullanılmaktadır. YZ destekli yaklaşımlar, sistemin çevresel değişkenlere gerçek zamanlı olarak uyum sağlamasına olanak tanır. Özellikle dijital ikiz teknolojilerinde, sistemlerin anlık verilerle güncellenerek daha doğru sonuçlar üretmesi sağlanmaktadır.

8.1.2 Simülasyonda Kullanılan Yapay Zekâ ve Makine Öğrenmesi Teknikleri

YZ ve MÖ'nün simülasyon süreçlerine entegrasyonu, çeşitli yöntemler aracılığıyla gerçekleştirilmektedir:

- Denetimli Öğrenme (Supervised Learning): Mevcut veri setleri kullanılarak, sistemlerin belirli girdilere karşı nasıl tepki vereceğini tahmin eden modeller geliştirilir.
- Denetimsiz Öğrenme (Unsupervised Learning): Veri kümeleri içindeki örüntüleri keşfetmek ve simülasyon sistemlerinde bilinmeyen yapıların belirlenmesine yardımcı olmak için kullanılır.

- Pekiştirmeli Öğrenme (Reinforcement Learning- RL): Karmaşık sistemlerin simülasyonunda sıkça başvurulan bir yaklaşımdır. Ödül-maliyet mekanizması ile çalışan RL algoritmaları, simülasyon ortamında optimal karar verme süreçlerini destekler.

- Genetik Algoritmalar ve Evrimsel Hesaplama: Optimizasyon problemlerinin çözümünde etkili olan bu yöntemler, simülasyon ortamlarında en iyi çözümleri bulmak için kullanılır.

- Sinir Ağları ve Derin Öğrenme: Karmaşık sistemlerin modellenmesi ve öngörülerde bulunulması amacıyla kullanılır. Özellikle büyük veri

kümelerine dayalı simülasyonlarda derin öğrenme teknikleri öne çıkmaktadır.

8.1.3 Yapay Zekâ Destekli Simülasyonun Uygulama Alanları

YZ ve MÖ tabanlı simülasyon yöntemleri, birçok farklı alanda geniş bir uygulama yelpazesine sahiptir:

- Mühendislik ve Üretim: Üretim süreçlerinin optimize edilmesi, tedarik zinciri yönetimi ve Endüstri 4.0 çerçevesinde akıllı fabrikaların modellenmesi.
- Sağlık: Hastalıkların yayılımının modellenmesi, ilaç geliştirme süreçleri ve biyomedikal sistemlerin simülasyonu.
- Finans: Piyasa hareketlerinin tahmin edilmesi, risk analizi ve algoritmik ticaret stratejilerinin geliştirilmesi.
- Akıllı Ulaşım ve Şehircilik: Otonom araç simülasyonları, trafik yönetimi ve akıllı şehir planlaması.
- Savunma ve Havacılık: Askeri stratejilerin modellenmesi, uçuş simülasyonları ve hava savunma sistemlerinin optimize edilmesi.

8.1.4 Gelecek Perspektifi ve Karşılaşılan Zorluklar

YZ ve MÖ'nün simülasyona entegrasyonu önemli avantajlar sunarken, bazı teknik ve etik zorlukları da beraberinde getirmektedir:

Veri Güvenliği ve Gizlilik: Simülasyonlarda kullanılan büyük veri setlerinin gizliliği ve güvenliği büyük bir sorun teşkil etmektedir. Özellikle sağlık, finans ve savunma gibi alanlarda veri mahremiyeti kritik bir konudur.

Açıklanabilirlik ve Güvenilirlik: YZ tabanlı simülasyonların karar verme süreçlerinde nasıl işlediğinin anlaşılması, model güvenilirliği açısından son derece önemlidir. Derin öğrenme modellerinin "kara kutu" yapısı, bu konuda önemli bir sınırlama oluşturmaktadır.

Hesaplama Maliyetleri: Yüksek doğrulukta YZ destekli simülasyonlar, büyük hesaplama gücü gerektirebilir. Bu durum, özellikle küçük ölçekli araştırmalar için maliyetli hale gelebilir.

Etik Sorunlar: Yapay zekânın simülasyon süreçlerinde kullanımı, kararların insan denetiminden çıkmasına yol açabilir. Özellikle otonom sistemlerde etik karar mekanizmalarının nasıl oluşturulacağı önemli bir tartışma konusudur.

YZ ve MÖ'nün simülasyon alanındaki rolü giderek genişlemekte olup, gelecekte bu alandaki entegrasyonun daha da derinleşeceği öngörülmektedir. Yapay zekâ tabanlı simülasyon sistemlerinin daha şeffaf, güvenilir ve sürdürülebilir hale getirilmesi, önümüzdeki yıllarda bu alandaki temel araştırma konularından biri olacaktır.

8.2 Dijital İkizler ve Endüstri 4.0

Endüstri 4.0 ile birlikte üretim, lojistik, sağlık, enerji ve birçok sektörde dijitalleşme süreçleri hız kazanmıştır. Bu dönüşümün merkezinde yer alan teknolojilerden biri dijital ikizlerdir. Dijital ikiz (Digital Twin) kavramı, fiziksel bir sistemin, nesnenin veya sürecin sanal bir temsilini oluşturmayı ve bu model üzerinden analizler yapmayı ifade eder. Bu teknoloji, gerçek dünya ile sanal dünya arasında çift yönlü bir etkileşim sağlayarak, karar alma süreçlerinde önemli avantajlar sunmaktadır. Dijital ikizler, veri analitiği, yapay zekâ, büyük veri ve simülasyon teknikleri ile entegre çalışarak fiziksel sistemlerin operasyonel süreçlerini iyileştirme ve öngörücü bakım gibi alanlarda büyük kolaylıklar sağlamaktadır. Bu bölümde, dijital ikizlerin temel prensipleri, Endüstri 4.0 ile ilişkisi, uygulama alanları ve karşılaşılan zorluklar ele alınacaktır.

8.2.1 Dijital İkiz Kavramı ve Temel Prensipleri

Dijital ikizler, fiziksel ve sanal sistemler arasında bir köprü oluşturarak gerçek dünyadaki süreçleri sanal ortamda modelleme ve optimize etme imkanı sunar. Dijital ikiz sistemleri esasen üç ana bileşenden oluşur:

1. Fiziksel Sistem: Sensörler, makineler, üretim hatları ve insan hareketleri gibi fiziksel unsurlardan meydana gelir.

2. Dijital Model: Fiziksel sistemin sanal bir kopyasıdır. Bu model, matematiksel formüller, yapay zekâ algoritmaları ve büyük veri analizleri ile desteklenir.

3. Bağlantı ve Veri Akışı: Fiziksel ve dijital sistemler arasında sürekli bir veri akışı sağlar. Nesnelerin İnterneti (IoT), bulut bilişim ve yüksek hızlı iletişim teknolojileri bu bağlantıyı mümkün kılar.

Dijital ikizler, gerçek zamanlı veri toplama, veri analizi, simülasyonlar aracılığıyla karar destek mekanizmaları oluşturma ve öngörücü bakım gibi işlevleri yerine getirerek sistemlerin daha verimli çalışmasına katkıda bulunur.

8.2.2 Endüstri 4.0 ile Dijital İkizlerin Entegrasyonu

Endüstri 4.0, üretim ve hizmet sektörlerinde dijital dönüşümü hızlandıran bir devrimdir. Bu süreçte, akıllı fabrikalar, otomasyon sistemleri ve veri odaklı üretim modelleri öne çıkmaktadır. Dijital ikizler, Endüstri 4.0'ın temel bileşenlerinden biri olarak şu avantajları sunar:

- Gerçek Zamanlı İzleme: Üretim süreçlerinin anlık takibini sağlar.
- Öngörücü Bakım: Makine arızalarını tahmin ederek bakım süreçlerini optimize eder.
- Süreç Optimizasyonu: Simülasyonlar sayesinde üretim hatlarını en verimli hale getirir.
- Kaynak Yönetimi: Malzeme, enerji ve iş gücü kullanımını en iyi şekilde planlamaya yardımcı olur.
- Ürün Yaşam Döngüsü Yönetimi: Yeni ürünlerin geliştirilme süreçlerini hızlandırır ve maliyetleri düşürür.

Dijital ikizlerin Endüstri 4.0 ile entegrasyonu, üretim süreçlerini daha verimli hale getirmenin yanı sıra sürdürülebilirlik ve çevresel etkilerin azaltılması konularında da önemli katkılar sağlar.

8.2.3 Dijital İkizlerin Uygulama Alanları

Dijital ikiz teknolojisi, birçok sektörde yaygın bir şekilde kullanılmaktadır:

- Üretim ve Otomotiv: Akıllı fabrikaların yönetimi, üretim süreçlerinin optimize edilmesi ve otomotiv sektöründe yeni nesil araç tasarımı.
- Sağlık: Cerrahi simülasyonlar, hasta takibi ve kişiye özel tedavi planlarının geliştirilmesi.

- Enerji: Akıllı şebekeler, enerji tüketim tahminleri ve yenilenebilir enerji yönetimi.

- Şehircilik: Akıllı şehir uygulamaları, trafik yönetimi ve altyapı planlaması.

- Havacılık ve Savunma: Uçak bakım süreçlerinin simülasyonu, otonom sistemlerin test edilmesi ve askeri stratejilerin modellenmesi.

Bu geniş uygulama alanları, dijital ikizlerin sistemlerin güvenilirliğini artırma, maliyetleri düşürme ve operasyonel verimliliği en üst düzeye çıkarma konularında önemli bir rol oynadığını göstermektedir.

8.2.4 Dijital İkizlerin Karşılaştığı Zorluklar ve Gelecek Perspektifi

Dijital ikiz teknolojisinin sağladığı avantajlara rağmen, bu sistemlerin geniş çapta uygulanmasında bazı teknik ve etik zorluklar mevcuttur:

1. Veri Güvenliği ve Gizlilik: Gerçek zamanlı veri toplayan dijital ikizlerin güvenliği son derece önemlidir. Siber saldırılara karşı etkili güvenlik önlemleri alınması gerekmektedir.

2. Yüksek Hesaplama Maliyeti: Gerçek zamanlı veri işleme ve karmaşık simülasyonlar, yüksek işlem gücü gerektirdiğinden maliyetli olabilir.

3. Standartlaşma Eksikliği: Dijital ikiz sistemlerinin farklı sektörlerde standart bir yapıya sahip olmaması, entegrasyon süreçlerini zorlaştırmaktadır.

4. Veri Doğruluğu ve Güvenilirliği: Simülasyon süreçlerinin başarılı olabilmesi için kullanılan verilerin güncel ve güvenilir olması şarttır. Yanlış veriler, hatalı analiz sonuçlarına yol açabilir.

5. Etik ve Sosyal Etkiler: Dijital ikizlerin insan hayatına etkisi, iş gücünde otomasyonun artışı gibi konular etik açıdan ele alınmalıdır.

Gelecekte, dijital ikizlerin daha akıllı, ölçeklenebilir ve otonom sistemler haline gelmesi beklenmektedir. Özellikle yapay zekâ ve makine öğrenmesi ile entegrasyonları sayesinde, bu sistemlerin daha hızlı, esnek ve öngörülebilir hale gelmesi öngörülmektedir. Ayrıca, çevresel

sürdürülebilirlik odaklı uygulamalarla, enerji verimliliğini artıran ve karbon ayak izini azaltan çözümler sunulması beklenmektedir.

Sonuç olarak, dijital ikiz teknolojisi, Endüstri 4.0'ın en önemli bileşenlerinden biri olarak öne çıkmakta ve gelecekte daha yaygın bir şekilde benimsenerek üretim, hizmet ve karar destek süreçlerinde devrim yaratması beklenmektedir.

8.3 Multidisipliner Simülasyonun Geleceği

Simülasyon teknikleri, mühendislik, tıp, ekonomi, sosyal bilimler ve daha birçok alanda karar destek sistemlerinin vazgeçilmez bir parçası haline gelmiştir. Günümüzde disiplinler arası araştırmaların artmasıyla birlikte multidisipliner simülasyon kavramı daha da önem kazanmıştır. Bu tür simülasyon, farklı bilimsel disiplinlerin simülasyon modellerinin bir araya getirilmesiyle daha kapsamlı ve gerçekçi analizler yapmayı hedeflemektedir. Bu yaklaşım, karmaşık sistemleri daha iyi anlamak, optimize etmek ve öngörülebilirlik sağlamak açısından kritik bir rol oynamaktadır. Bu bölümde, multidisipliner simülasyonun temel bileşenleri, avantajları, uygulama alanları ve gelecekte karşılaşılabilecek zorluklar üzerinde durulacaktır.

8.3.1 Multidisipliner Simülasyonun Tanımı ve Temel İlkeleri

Multidisipliner simülasyon, birden fazla disiplinin modelleme ve simülasyon tekniklerini bir araya getirerek daha kapsamlı bir analiz ortamı oluşturmayı amaçlayan bir yaklaşımdır. Geleneksel simülasyon yöntemleri genellikle tek bir disipline odaklanırken, multidisipliner simülasyon sistemleri karmaşık, dinamik ve çok yönlü süreçleri ele alarak daha gerçekçi sonuçlar sunar.

Multidisipliner simülasyonun temel ilkeleri şunlardır:

- **Disiplinler Arası Entegrasyon:** Mühendislik, biyoloji, ekonomi, lojistik, sağlık ve diğer alanlardaki simülasyon modellerinin etkileşimli bir şekilde çalışmasını sağlar.
- **Modülerlik ve Uyarlanabilirlik:** Modellerin birbirine entegre edilebilir şekilde tasarlanması ve farklı disiplinlere kolayca uygulanabilmesi önemlidir.

- Gerçek Zamanlı Veri Kullanımı: Büyük veri analitiği, yapay zekâ ve nesnelerin interneti (IoT) gibi teknolojilerle desteklenerek güncel ve doğru simülasyon çıktıları elde edilir.

- Doğrulama ve Geçerlilik: Farklı disiplinlerden gelen verilerin tutarlılığı, model doğrulama süreçleriyle sağlanmalıdır.

Bu ilkeler doğrultusunda geliştirilen multidisipliner simülasyon sistemleri, disiplinler arası iş birliğini teşvik ederek daha doğru ve etkili karar alma mekanizmaları oluşturur.

8.3.2 Multidisipliner Simülasyonun Avantajları

Multidisipliner simülasyon, geleneksel tek disiplinli simülasyon yaklaşımlarına göre birçok avantaj sunmaktadır:

1. Gerçekçi ve Kapsamlı Modeller: Gerçek dünyadaki sistemler genellikle birçok değişken ve disiplini içerdiğinden, multidisipliner simülasyonlar daha doğru tahminler yapma imkanı tanır.

2. Risk ve Maliyet Azaltımı: Farklı sektörlerde karar alma süreçlerinde kullanılabilen bu simülasyonlar, potansiyel hataları önceden tespit ederek zaman ve maliyet açısından önemli tasarruf sağlar.

3. Karar Destek Mekanizmalarının Güçlenmesi: Farklı disiplinlerin bakış açılarını bir araya getirerek daha sağlıklı ve bilinçli kararlar alınmasına yardımcı olur.

4. Esneklik ve Uyarlanabilirlik: Gelişen teknolojilerle sürekli güncellenebilen ve farklı alanlara uyum sağlayabilen modeller oluşturulabilir.

5. Öngörü Yeteneğinin Artması: Büyük veri ve yapay zeka destekli multidisipliner simülasyonlar, gelecekteki senaryoları analiz etme konusunda büyük avantaj sunar.

Bu avantajlar, multidisipliner simülasyonları birçok sektörde vazgeçilmez bir araç haline getirmektedir.

8.3.3 Multidisipliner Simülasyonun Uygulama Alanları

Multidisipliner simülasyon, geniş bir uygulama yelpazesine sahiptir ve birçok sektörde etkin bir şekilde kullanılmaktadır:

- **Sağlık ve Biyomedikal Simülasyonlar:** Hasta takibi, hastalık modelleme, ilaç geliştirme ve cerrahi planlama gibi alanlarda önemli bir rol oynamaktadır. Örneğin, tıp ve mühendislik disiplinlerinin birleşimiyle geliştirilen biyomedikal simülasyonlar, kişiye özel tedavi yöntemleri sunma konusunda büyük katkılar sağlamaktadır.

- **Endüstriyel Üretim ve Lojistik:** Tedarik zinciri yönetimi, üretim süreçlerinin optimizasyonu ve robotik sistemlerin entegrasyonu gibi konularda multidisipliner simülasyonlar yaygın bir şekilde kullanılmaktadır.

- **Çevresel ve İklim Modelleri:** Atmosfer, su kaynakları ve enerji sistemleri gibi alanlarda multidisipliner simülasyonlar, sürdürülebilir çözümler geliştirmek için önemli bir araçtır.

- **Havacılık ve Savunma:** Uçak tasarımı, savaş simülasyonları ve insansız hava araçları (İHA) gibi alanlarda farklı mühendislik disiplinlerinin bir araya gelmesiyle simülasyon modelleri oluşturulmaktadır.

- **Akıllı Şehirler ve Ulaşım Sistemleri:** Trafik yönetimi, enerji dağıtımı ve altyapı planlaması gibi konularda multidisipliner simülasyonlar, daha verimli ve sürdürülebilir çözümler geliştirilmesine yardımcı olmaktadır.

Bu uygulama alanları, multidisipliner simülasyonların sadece akademik bir araştırma aracı olmanın ötesinde, endüstriyel ve toplumsal problemlerin çözümünde de kritik bir rol oynadığını göstermektedir.

8.3.4 Multidisipliner Simülasyonun Geleceği ve Karşılaşılan Zorluklar

Gelecekte multidisipliner simülasyonların daha yaygın ve etkili bir şekilde kullanılabilmesi için bazı temel zorlukların üstesinden gelinmesi gerekmektedir:

1. Model Entegrasyonu: Farklı disiplinlerden gelen simülasyon modellerinin uyumlu bir şekilde çalışabilmesi için ortak standartların geliştirilmesi önemlidir.

2. Hesaplama Gücü ve Veri Yönetimi: Gerçek zamanlı simülasyonların giderek daha karmaşık hale gelmesi, yüksek işlem gücü gereksinimlerini artırmaktadır. Bulut bilişim ve kuantum hesaplama gibi teknolojiler, bu sorunun çözümünde potansiyel fırsatlar sunmaktadır.

3. Doğruluk ve Güvenilirlik: Multidisipliner modellerin güvenilir tahminler yapabilmesi için veri doğruluğu ve model kalibrasyonu kritik öneme sahiptir.

4. Etik ve Yasal Sorunlar: Özellikle sağlık ve otonom sistemler gibi alanlarda multidisipliner simülasyonların etik boyutları ve veri gizliliği gibi konuların dikkate alınması gerekmektedir.

Gelecekte yapay zeka, büyük veri, kuantum hesaplama ve nesnelerin interneti gibi teknolojilerin gelişimiyle birlikte, multidisipliner simülasyonlar daha da güçlenecek ve geniş bir uygulama alanı bulacaktır. Özellikle özerk sistemler, akıllı şehirler, biyoteknoloji ve sürdürülebilirlik alanlarında bu teknolojinin kullanımı giderek artacaktır.

Sonuç olarak, multidisipliner simülasyon, geleceğin karmaşık problemlerinin çözümünde vazgeçilmez bir araç haline gelecek ve farklı disiplinlerin iş birliğiyle daha kapsamlı, doğru ve etkili modellerin geliştirilmesine katkıda bulunacaktır.

8.4 Kuantum Bilişim ve Simülasyon

Geleneksel hesaplama yöntemleri, karmaşık sistemlerin modellenmesi ve simülasyonu konusunda belirli sınırlamalarla karşılaşmaktadır. Özellikle yüksek boyutlu problemlerin çözümü, büyük veri analizi ve kuantum mekaniğine dayalı sistemlerin simülasyonu, klasik bilgisayarlarla zaman açısından maliyetli veya pratik olarak imkânsız hale gelebilir. Kuantum bilişim, bu sınırlamaları aşmak için süperpozisyon, dolanıklık ve kuantum geçişkenlik gibi fiziksel prensiplere dayanan hesaplama yöntemleri sunarak simülasyon alanında devrim yaratma potansiyeline sahiptir. Bu bölümde, kuantum bilişimin simülasyon

teknikleri üzerindeki etkisi, avantajları, uygulama alanları ve karşılaşılan zorluklar ele alınacaktır.

8.4.1 Kuantum Bilişimin Temel İlkeleri ve Simülasyon ile İlişkisi

Kuantum bilişim, klasik hesaplama yöntemlerinden farklı olarak kuantum mekaniği prensiplerine dayanan bir hesaplama biçimidir. Bu bağlamda, kuantum bilişimi klasik simülasyon yöntemlerinden ayıran temel unsurlar şunlardır:

- Süperpozisyon: Kuantum bitleri (qubit), 0 ve 1 durumlarını aynı anda barındırarak paralel hesaplama imkanı sunar.
- Dolanıklık: Birden fazla qubit'in birbirine bağımlı hale gelmesi, uzak mesafelerde bile anında bilgi alışverişine olanak tanır.
- Kuantum Girişim: Kuantum durumlarının süperpozisyon etkisi, belirli hesaplamaların daha hızlı bir şekilde yapılmasını sağlar.

Simülasyon açısından değerlendirildiğinde, kuantum bilgisayarlar özellikle doğrusal olmayan dinamik sistemler, istatistiksel analizler ve yüksek boyutlu optimizasyon problemleri gibi alanlarda devrim niteliğinde çözümler sunabilir.

8.4.2 Kuantum Simülasyonun Avantajları

Kuantum bilgisayarların simülasyon alanına katkıları birçok açıdan değerlendirilebilir:

1. Karmaşık Sistemlerin Daha Hızlı ve Hassas Simülasyonu: Moleküler dinamikler, kimyasal reaksiyonlar ve fiziksel süreçlerin modellenmesi, klasik bilgisayarlar için büyük işlem gücü gerektirirken, kuantum simülasyonlar bu süreçleri çok daha hızlı ve hassas bir şekilde analiz edebilir.

2. Yüksek Boyutlu Optimizasyon Problemlerinin Çözülmesi: Kuantum bilişim, büyük ölçekli lojistik, finansal risk analizi ve tedarik zinciri optimizasyonu gibi alanlarda karmaşık karar süreçlerini hızlandırma potansiyeline sahiptir.

3. Monte Carlo Simülasyonlarının Güçlenmesi: Finans, biyoloji ve mühendislik gibi alanlarda yaygın olarak kullanılan Monte Carlo

simülasyonları, kuantum algoritmaları sayesinde daha hızlı ve daha az hata ile gerçekleştirilebilir.

4. Doğal Kuantum Sistemlerinin Modellenmesi: Kuantum fiziğine dayalı sistemler, klasik bilgisayarların simüle etmede zorlandığı fenomenleri daha gerçekçi bir şekilde modelleme imkânı sunar.

Bu avantajlar, kuantum bilişimin simülasyon alanındaki mevcut sınırlamaları aşmada kritik bir rol oynayacağını göstermektedir.

8.4.3 Kuantum Simülasyonun Uygulama Alanları

Kuantum bilişim destekli simülasyonlar, birçok alanda devrim yaratma potansiyeline sahiptir:

- Kimya ve Malzeme Bilimi: Moleküler yapıların simülasyonu, yeni ilaç keşifleri, süperiletken malzemeler ve nano-teknoloji gibi konularda önemli ilerlemeler sağlanabilir.

- Finans ve Ekonomi: Kuantum hesaplama, portföy optimizasyonu, risk analizi ve piyasa dalgalanmalarının tahmini gibi finansal simülasyon süreçlerinde etkili bir şekilde kullanılabilir.

- İklim ve Çevre Bilimleri: Atmosfer modellerinin, iklim değişikliği senaryolarının ve enerji verimliliği simülasyonlarının geliştirilmesinde kuantum bilişim önemli katkılar sunabilir.

- Makine Öğrenmesi ve Yapay Zekâ: Kuantum bilişim destekli yapay zekâ algoritmaları, büyük veri analizi ve karmaşık karar destek sistemlerinde daha hızlı ve etkili çözümler sağlayabilir.

- Biyolojik Simülasyonlar: Protein katlanma süreçleri, genetik analizler ve biyomedikal modellemeler gibi karmaşık biyolojik süreçlerin simülasyonu kuantum bilgisayarlarla daha verimli bir şekilde gerçekleştirilebilir.

Bu alanlardaki gelişmeler, kuantum simülasyonların yalnızca teorik bir yenilik olmaktan çıkıp pratik uygulamalarda büyük bir etki yaratacağını göstermektedir.

8.4.4 Kuantum Simülasyonun Geleceği ve Karşılaşılan Zorluklar

Kuantum bilişim ve simülasyon alanındaki gelişmeler umut verici olsa da, bu teknolojinin yaygınlaşmasının önünde bazı temel engeller bulunmaktadır:

1. Donanım Sınırlamaları: Günümüzde kuantum bilgisayarların ölçeklenebilirliği, hata oranları ve hesaplama stabilitesi gibi teknik zorluklar hâlâ çözülmeyi bekliyor.

2. Klasik ve Kuantum Algoritmalarının Entegrasyonu: Mevcut simülasyon yöntemleri ile kuantum tabanlı yaklaşımların nasıl bir arada çalışabileceği konusunda daha fazla araştırma yapılması gerekiyor.

3. Hata Düzeltme ve Dayanıklılık: Kuantum bilgisayarlar, klasik bilgisayarlara göre daha hassas olup dış etkenlerden (örneğin, manyetik alanlar ve sıcaklık değişimleri) kolayca etkilenabiliyor.

4. Veri Güvenliği ve Kuantum Kriptografi: Kuantum bilgisayarların gelişimiyle birlikte şifreleme yöntemleri ve veri güvenliği konularında yeni çözümler geliştirilmesi gerekiyor.

Gelecekte, kuantum bilişim teknolojisindeki ilerlemeler simülasyon süreçlerinde devrim yaratacak ve özellikle büyük ölçekli sistemlerin modellenmesi konusunda yeni ufuklar açacaktır. Özellikle hata düzeltme mekanizmaları, hibrit kuantum-klasik algoritmalar ve ölçeklenebilir kuantum donanımları konusundaki gelişmeler, bu teknolojinin endüstriyel ve akademik kullanımlarını hızlandıracaktır. Sonuç olarak, kuantum bilişim ve simülasyon tekniklerinin entegrasyonu, multidisipliner araştırmaların ve karar destek sistemlerinin yeni bir seviyeye ulaşmasını sağlayarak, geleceğin bilimsel ve teknolojik devrimlerinden biri olma potansiyeline sahiptir.

8.5 Etik ve Sürdürülebilirlik

Simülasyon teknolojileri, karar verme süreçlerini geliştirme, kaynak kullanımını optimize etme ve karmaşık sistemleri anlama konusunda etkili araçlar sunmaktadır. Ancak bu teknolojilerin yaygınlaşmasıyla birlikte etik ve sürdürülebilirlik meseleleri daha fazla önem kazanmaktadır. Simülasyon sistemlerinin geliştirilmesi ve uygulanması, yalnızca teknik verimlilik açısından değil, aynı zamanda toplumsal, çevresel ve etik

boyutlarıyla da ele alınmalıdır. Bu bölümde, simülasyon çalışmalarında etik ilkeler, çevresel sürdürülebilirlik ve sosyal sorumluluk kavramları çerçevesinde dikkate alınması gereken temel noktalar incelenecektir.

8.5.1 Simülasyonda Etik İlkeler

Simülasyon sistemlerinin geliştirilmesi ve kullanımı sırasında etik sorumlulukların dikkate alınması önemlidir. Etik ilkeler genellikle aşağıdaki temel unsurları içerir:

- **Şeffaflık ve Hesap Verebilirlik:** Simülasyon sonuçlarının nasıl elde edildiğinin açık bir şekilde belirtilmesi ve kullanılan veri kaynaklarının güvenilir olması, bilimsel ve akademik çalışmaların temel gerekliliklerindendir.

- **Yanlılık ve Tarafsızlık:** Simülasyon modellerinin geliştirilmesi ve sonuçlarının yorumlanmasında bilinçli veya bilinçsiz yanlılıkların önlenmesi büyük önem taşır. Veri seçimi, modelleme yöntemleri ve parametrelerin belirlenmesi aşamalarında nesnelliğin korunması gerekmektedir.

- **Gizlilik ve Veri Güvenliği:** Simülasyon çalışmaları genellikle büyük ölçekli veri setleri içerir ve bu durum, bireylerin veya kurumların mahremiyet haklarını ihlal etme riski doğurabilir. Kişisel ve hassas bilgilerin korunması için etik veri yönetimi uygulamalarının benimsenmesi şarttır.

- **Toplumsal Etki ve Sorumluluk:** Simülasyon sistemlerinin karar destek süreçlerinde kullanımı, bireyler ve toplumlar üzerinde doğrudan veya dolaylı etkiler yaratabilir. Bu nedenle, uygulamaların sosyal adalet, eşitlik ve kamu yararına uygun olması sağlanmalıdır.

Etik sorunların göz ardı edilmesi, yanlış yönlendirilmiş politikalar ve toplumsal güven kaybı gibi ciddi sonuçlara yol açabilir. Bu nedenle, simülasyon süreçlerinin etik değerlendirmeye tabi tutulması, güvenilirliklerini ve toplumsal kabul edilebilirliklerini artıracaktır.

8.5.2 Sürdürülebilir Simülasyon ve Çevresel Etkiler

Simülasyon teknolojileri, çevresel sürdürülebilirliği destekleyen önemli bir araç olarak öne çıkmaktadır. Özellikle iklim değişikliği, enerji yönetimi, şehir planlaması ve doğal kaynakların korunması gibi alanlarda simülasyon kullanımı büyük avantajlar sunmaktadır. Ancak, simülasyonların kendilerinin de çevresel etkileri dikkate alınmalıdır.

- Enerji Tüketimi ve Karbon Ayak İzi: Yüksek performanslı hesaplama sistemleri ve büyük ölçekli veri merkezleri, yoğun enerji tüketimi gerektirir. Yeşil bilişim uygulamalarının benimsenmesi, karbon ayak izinin azaltılması açısından kritik bir rol oynamaktadır.

- Doğal Kaynak Yönetimi: Simülasyonlar, su yönetimi, tarımsal verimlilik ve sanayi süreçlerinde sürdürülebilir stratejilerin geliştirilmesine katkıda bulunabilir.

- İklim ve Çevre Modelleri: Simülasyon teknikleri, hava tahmini, iklim değişikliği senaryoları ve ekolojik sistem analizleri gibi konularda kullanılarak çevresel sürdürülebilirliğe katkı sağlayabilir.

- Atık Yönetimi ve Geri Dönüşüm: Üretim süreçlerinin simüle edilmesi, atık miktarının azaltılmasına ve döngüsel ekonominin teşvik edilmesine yardımcı olabilir.

Bu bağlamda, sürdürülebilir simülasyon modelleri geliştirilerek, hem çevresel etkiler en aza indirilebilir hem de doğaya duyarlı karar alma süreçleri desteklenebilir.

8.5.3 Sosyal Sürdürülebilirlik ve Simülasyon

Simülasyon sistemlerinin toplumsal fayda sağlamak amacıyla kullanılması, sosyal sürdürülebilirlik açısından büyük bir öneme sahiptir. Eğitim, sağlık, kentleşme ve kriz yönetimi gibi alanlarda simülasyon tabanlı yaklaşımlar, toplumsal refahı artırmaya yönelik etkili çözümler sunabilir.

- Eğitimde Simülasyon: Simülasyon teknikleri, öğrencilere etkileşimli ve deneyimsel öğrenme fırsatları sunarak eğitim süreçlerini geliştirebilir.

- Sağlık Hizmetlerinde Simülasyon: Tıbbi simülasyonlar, cerrahi planlamadan salgın hastalık modellemelerine kadar birçok alanda sağlık hizmetlerinin kalitesini artırma potansiyeline sahiptir.

- Afet ve Kriz Yönetimi: Deprem, sel ve diğer doğal afetler için gerçekleştirilen simülasyon çalışmaları, hızlı müdahale süreçlerinin iyileştirilmesine ve risk azaltma stratejilerinin geliştirilmesine yardımcı olur.

- Kentleşme ve Altyapı Planlaması: Akıllı şehir uygulamaları, trafik yönetimi, enerji tüketimi ve kamu hizmetleri gibi alanlarda simülasyon destekli karar verme mekanizmalarının oluşturulmasına olanak tanır.

Bu noktada, simülasyonun yalnızca teknik bir araç olarak değil, toplumsal refahı artıran bir mekanizma olarak değerlendirilmesi gerektiği vurgulanmalıdır.

8.5.4 Etik ve Sürdürülebilir Simülasyon İçin Gelecek Perspektifleri

Gelecekte, simülasyon teknolojilerinin etik ve sürdürülebilirlik ilkeleri doğrultusunda daha dikkatli bir şekilde geliştirilmesi ve uygulanması gerekecektir. Bu bağlamda aşağıdaki konular öne çıkacaktır:

- Şeffaf ve Açık Kaynak Simülasyon Sistemleri: Simülasyon süreçlerinin daha erişilebilir ve denetlenebilir hale gelmesi için açık kaynak yazılımların yaygınlaştırılması büyük önem taşımaktadır.

- Sürdürülebilir Bilişim Teknolojileri: Simülasyon altyapılarının daha az enerji tüketen ve çevre dostu veri merkezleriyle desteklenmesi gerekmektedir.

- Hukuki ve Regülasyonel Çerçeveler: Simülasyon teknolojilerinin etik kullanımı için uluslararası düzenlemelerin ve standartların geliştirilmesi önemlidir.

- Toplumsal Farkındalık ve Eğitim: Etik simülasyon ve sürdürülebilirlik konularında farkındalık yaratılması, kullanıcıların bilinçli hareket etmelerini sağlayacaktır.

Sonuç olarak, etik ve sürdürülebilirlik ilkeleri çerçevesinde geliştirilen simülasyon sistemleri, hem bilimsel araştırmalarda hem de sanayi ve kamu

politikalarında daha sağlıklı kararlar alınmasına yardımcı olacaktır. Simülasyonların adil, şeffaf ve çevresel etkileri dikkate alan bir anlayışla kullanılması, gelecekte daha güvenilir ve etkili modellerin oluşturulmasını mümkün kılacaktır.

Simülasyon teknolojileri, disiplinler arası etkileşimlerin artması, dijitalleşmenin hız kazanması ve hesaplama gücünün gelişmesiyle birlikte giderek daha büyük bir öneme sahip olmaktadır. Yapay zekâ, makine öğrenmesi, dijital ikizler ve kuantum bilişim gibi alanlarla birleşen simülasyon yaklaşımları, karar verme süreçlerinde devrim yaratmakta ve geleneksel modelleme yöntemlerini aşan yenilikçi çözümler sunmaktadır.

Bu kitapta ele alınan yöntemler, simülasyonun yalnızca teknik bir araç olmanın ötesine geçtiğini, karmaşık sistemleri anlamada, sürdürülebilir politikalar geliştirmede ve çok boyutlu problemlere etkili çözümler üretmede kritik bir rol oynadığını ortaya koymaktadır. Özellikle disiplinler arası uygulamalar, simülasyon teknolojilerinin geleceğini şekillendirecek en önemli faktörlerden biridir. Mühendislikten sağlık bilimine, ekonomiden şehir planlamasına kadar geniş bir yelpazede simülasyon tekniklerinin kullanımının artması, bu alandaki araştırmaları daha kapsayıcı ve uygulanabilir hale getirecektir.

Öte yandan, simülasyonların etik ve sürdürülebilirlik çerçevesinde ele alınması, teknolojinin sadece verimlilik odaklı değil, toplumsal fayda sağlayan ve çevresel sorumlulukları gözetken bir anlayışla kullanılması açısından büyük önem taşımaktadır. Şeffaf veri yönetimi, tarafsız modelleme yaklaşımları ve sürdürülebilir hesaplama altyapıları, gelecek nesillerin daha güvenilir ve etkili simülasyon modellerine erişimini sağlayacaktır.

Gelecekte simülasyon teknolojilerinin yönü, etik kurallar, yapay zekâ destekli modeller, dijital ikiz uygulamaları ve kuantum hesaplama teknikleriyle şekillenecektir. Ancak bu gelişmelerin başarısı, yalnızca teknik ilerlemelere değil, aynı zamanda bilim insanlarının, karar vericilerin ve toplumun bu teknolojilere olan yaklaşımına da bağlıdır.

Sonuç olarak, simülasyon sistemleri, teknolojik dönüşümün en kritik bileşenlerinden biri olarak kalmaya devam edecektir. Bu kitapta ele alınan konuların, okuyuculara simülasyonun geleceğine dair kapsamlı bir bakış

açısı sunduğunu ve bu alandaki çalışmalara yön vermede katkı sağladığını umuyoruz.

Son Söz

Bu kitap, simülasyon ve matematiksel modelleme dünyasında yol almak isteyenler için kapsamlı bir rehber niteliğinde hazırlanmıştır. Amacımız, farklı disiplinlerden gelen araştırmacıların ve öğrencilerin, simülasyon tekniklerini daha bilinçli bir şekilde seçmesini ve uygulamasını sağlamaktır.

Günümüzde simülasyon yalnızca teknik bir araç olmaktan çıkıp, karar destek sistemlerinden sürdürülebilirlik çalışmalarına kadar geniş bir etki alanına sahiptir. Bu bağlamda, kitabın sunduğu yöntemler ve perspektifler, okurların hem teorik bilgiyi kavramalarına hem de pratik uygulamalar geliştirmelerine yardımcı olmayı hedeflemektedir.

Ancak simülasyonun gücü yalnızca kullanılan modellerle değil, bu modellerin nasıl ve ne amaçla kullanıldığıyla belirlenir. Doğru yöntemleri seçmek, verileri etik çerçevede değerlendirmek ve disiplinler arası iş birliklerine açık olmak, simülasyon çalışmalarının başarıya ulaşmasında kritik rol oynayacaktır.

Bu kitabın, simülasyon ve modelleme alanında çalışan herkes için bir başlangıç noktası, bir rehber ve ilham kaynağı olmasını diliyoruz. Bilim ve teknoloji her geçen gün gelişirken, simülasyonun sunduğu fırsatlar da genişlemeye devam edecektir. Önemli olan, bu fırsatları bilinçli ve sorumlu bir şekilde kullanmaktır.

Simülasyon dünyasında başarılar dilerim!

Kaynak

- [1] Banks, J., Carson, J. S. II, & Nelson, B. C. (1996). Discrete-event system simulation (2nd ed.). Prentice Hall.
- [2] Law, A. M., & Kelton, W. D. (1991). Simulation modeling and analysis (2nd ed.). McGraw-Hill.
- [3] Pritsker, A. A. B., Elliott Sigal, C. E., & Hammesfahr, R. D. J. (1989). SLAM II network models for decision support. Prentice Hall.
- [4] Pritsker, A. A. B. (1995). Introduction to simulation and SLAM II (4th ed.). Wiley.
- [5] Pritsker Corporation. (1993). Simulation: A decision support tool. Pritsker Corporation.
- [6] Musselman, K. J. (1992). Conducting a successful simulation project. In J. J. Swain, D. Goldsman, R. C. Crain, & J. R. Wilson (Eds.), Proceedings of the 1992 Winter Simulation Conference (pp. 115-121). IEEE.
- [7] Musselman, K. J. (1993). Guidelines for simulation project success. In G. W. Evans, M. Mollaghaserni, E. G. Russell, & W. E. Biles (Eds.), Proceedings of the 1993 Winter Simulation Conference (pp. 58-64). IEEE.
- [8] Fishman, G. S. (2001). Discrete-event simulation: Modeling, programming, and analysis. Springer.
- [9] Shannon, R. E. (1975). Systems simulation: The art and science. Prentice Hall.
- [10] Banks, J., Buckley, S., Jain, S., Lendermann, P., & Manivannan, M. S. (2002). Panel session: Opportunities for simulation in supply chain management. Proceedings of the Winter Simulation Conference, 1652–1658.
- [11] Kelton, W. D., Sadowski, R. P., & Sturrock, D. T. (2010). Simulation with Arena (5th ed.). McGraw-Hill.
- [12] Ross, S. M. (2013). Introduction to probability models (11th ed.). Academic Press.
- [13] Sargent, R. G. (2011). Verification and validation of simulation models. Journal of Simulation, 7(1), 12–24.

- [14] Borshchev, A. (2013). *The Big Book of Simulation Modeling: Multimethod Modeling with AnyLogic 6*. AnyLogic North America.
- [15] Sterman, J. D. (2000). *Business dynamics: Systems thinking and modeling for a complex world*. Irwin/McGraw-Hill.
- [16] Law, A. M. (2015). *Simulation modeling and analysis* (5th ed.). McGraw-Hill.
- [17] Seila, A. F., Ceric, V., & Tadikamalla, P. (2003). *Applied simulation modeling*. Thomson Learning.
- [18] Fishwick, P. A. (1995). *Simulation model design and execution: Building digital worlds*. Prentice Hall.
- [19] Nance, R. E. (1993). A history of discrete event simulation programming languages. *ACM SIGPLAN Notices*, 28(3), 149–175.
- [20] Pidd, M. (2004). *Computer simulation in management science* (5th ed.). John Wiley & Sons.
- [21] Harrell, C. R., Ghosh, B. K., & Bowden, R. O. (2011). *Simulation using ProModel* (3rd ed.). McGraw-Hill.
- [22] Innis, G., & Rexstad, E. (1983). Simulation modeling methodology: Principles and applications. *The Journal of Wildlife Management*, 47(4), 1134–1142.
- [23] Page, E. H., & Smith, R. (1998). Introduction to military training simulation: A guide for discrete event simulationists. *Proceedings of the Winter Simulation Conference*, 53-60.
- [24] Ören, T. I., & Zeigler, B. P. (1999). Simulation methodologies. In A. P. Sage & W. B. Rouse (Eds.), *Handbook of Systems Engineering and Management* (pp. 847–888). Wiley.
- [25] Banks, J., & Gibson, R. R. (1997). Don't automate chaos: A guide to success in simulation. *Proceedings of the 1997 Winter Simulation Conference*, 49-55.
- [26] Borshchev, A., & Filippov, A. (2004). From system dynamics and discrete event to practical agent-based modeling: Reasons, techniques, tools. *Proceedings of the 22nd International Conference of the System Dynamics Society*.

- [27] Zeigler, B. P., Praehofer, H., & Kim, T. G. (2000). *Theory of modeling and simulation: Integrating discrete event and continuous complex dynamic systems* (2nd ed.). Academic Press.
- [28] Hollocks, B. W. (2006). Forty years of discrete-event simulation – A personal reflection. *Journal of the Operational Research Society*, 57(12), 1383–1399.
- [29] Liu, L., & Kärkkäinen, M. (2016). Simulation-based decision support for supply chain logistics: A review. *Computers & Industrial Engineering*, 102, 51–68.
- [30] Djanatljev, A., & German, R. (2013). Prospective healthcare decision-making by combined system dynamics, discrete-event, and agent-based simulation. *Proceedings of the Winter Simulation Conference*, 270–281.
- [31] Kleijnen, J. P. C. (2008). *Design and analysis of simulation experiments*. Springer Science & Business Media.
- [32] Balci, O. (1997). Verification, validation and accreditation of simulation models. *Proceedings of the 1997 Winter Simulation Conference*, 135–141.
- [33] Chwif, L., & Paul, R. J. (2000). On simulation model complexity. *Proceedings of the Winter Simulation Conference*, 449–455.
- [34] Martinez, J. C., & Ioannou, P. G. (1999). General-purpose simulation with Stroboscope. *Proceedings of the 1999 Winter Simulation Conference*, 1159–1168.
- [35] Jahangirian, M., Eldabi, T., Naseer, A., Stergioulas, L. K., & Young, T. (2010). Simulation in manufacturing and business: A review. *European Journal of Operational Research*, 203(1), 1–13.
- [36] Robinson, S. (2005). Discrete-event simulation: From the pioneers to the present, what next? *Journal of the Operational Research Society*, 56(6), 619–629.
- [37] Tako, A. A., & Robinson, S. (2012). The application of discrete event simulation and system dynamics in the logistics and supply chain context. *Decision Support Systems*, 52(4), 802–815.