

T. C.
MARMARA ÜNİVERSİTESİ
SOSYAL BİLİMLER ENSTİTÜSÜ
EKONOMETRİ ANABİLİM DALI
YÖNEYLEM ARAŞTIRMASI BİLİM DALI

**MONTE CARLO SİMÜLASYONU YAKLAŞIMIYLA KUYRUK
TEORİSİNİN İNCELENMESİ VE OTOMOTİV SEKTÖRÜ
ÜZERİNE BİR UYGULAMA**

Yüksek Lisans Tezi

MURAT USTAOĞLU

İstanbul, 2008

T. C.
MARMARA ÜNİVERSİTESİ
SOSYAL BİLİMLER ENSTİTÜSÜ
EKONOMETRİ ANABİLİM DALI
YÖNEYLEM ARAŞTIRMASI BİLİM DALI

**MONTE CARLO SİMÜLASYONU YAKLAŞIMIYLA KUYRUK
TEORİSİNİN İNCELENMESİ VE OTOMOTİV SEKTÖRÜ
ÜZERİNE BİR UYGULAMA**

Yüksek Lisans Tezi

MURAT USTAOĞLU

Danışman : Yrd. Doç. Dr. ERDAL DİNÇER

İstanbul, 2008

Marmara Üniversitesi
Sosyal Bilimler Enstitüsü Müdürlüğü

Tez Onay Belgesi

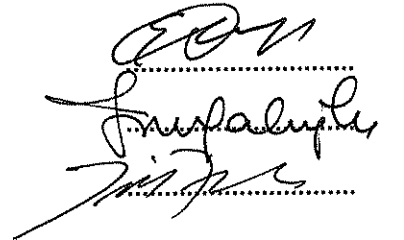
EKONOMETRİ Anabilim Dalı YÖNEYLEM ARAŞTIRMASI Bilim Dalı
Yüksek Lisans öğrencisi MURAT USTAOĞLU'nın MONTE CARLO SİMÜLASYONU
YAKLAŞIMIYLA KUYRUK TEORİSİNİN İNCELENMESİ VE OTOMOTİV SEKTÖRÜ
ÜZERİNE BİR UYGULAMA adlı tez çalışması, Enstitümüz Yönetim Kurulunun
19.06.2008 tarih ve 2008-10/25 sayılı kararıyla oluşturulan jüri tarafından
oybirliği/oyçokluğu ile Yüksek Lisans Tezi olarak kabul edilmiştir.

Öğretim Üyesi Adı Soyadı

İmzası

Tez Savunma Tarihi : 30.06.2008

- 1) Tez Danışmanı : YRD. DOÇ.DR. ERDAL DİNÇER
2) Jüri Üyesi : YRD. DOÇ.DR. FATMA URFALIOĞLU
3) Jüri Üyesi : DOÇ. DR. SİNAN ASLAN



GENEL BİLGİLER

Adı Ve Soyadı	: Murat USTAOĞLU
Ana Bilim Dalı	: Ekonometri
Bilim Dalı	: Yöneylem Araştırması
Tez Danışmanı	: Yrd. Doç. Dr. Erdal DİNÇER
Tez Türü Ve Tarihi	: Yüksek Lisans – Haziran 2008
Anahtar Kelimeler	: Kuyruk Teorisi, Kuyruk Modelleri, Monte Carlo Metodu

ÖZET

MONTE CARLO SİMÜLASYONU YAKLAŞIMIYLA KUYRUK TEORİSİNİN İNCELENMESİ VE OTOMOTİV SEKTÖRÜ ÜZERİNE BİR UYGULAMA

Son yıllarda artan nüfus ve insan ihtiyaçlarının mutlak ölçüde karşılanması ekonominin temel alanlarından biri olmuştur. İhtiyaçların karşılanabilmesi aşamasında üstün düzeyde ve tatminkar bir hizmet vererek müşteri memnuniyetini sağlamayı amaçlayan işletme sahipleri ile mal ve hizmet ihtiyacını karşılamak arzusuyla işletmelerden faydalanmayı amaçlayan müşteriler çok sık bir araya gelirler. Müşterilerin oluşan veya oluşabilecek kuyruklar karşısındaki tutumları (Kuyruğa girerek beklemek, hiç kuyruğa girmemek, kuyruğa girip bir süre sonra zaman kaybı sebebiyle vazgeçmek, vb.) araştırmansın konusunu oluşturmaktadır.

Monte Carlo metodu ile diğer yöneylem araştırması konularında olduğu gibi kuyruk modellerinde de rasgele dağılmış gözlemler kullanılarak denemeler yapılabilir. Bu sayede gerçekten deneme yaparak oluşabilecek maliyet ve zaman kaybının önüne geçilebilir.

GENERAL KNOWLEDGE

Name And Surname	: Murat USTAOĞLU
Field	: Econometrics
Programme	: Operations Research
Consultant	: Yrd. Doc. Dr. Erdal DINCER
Degree awarded And Date	: Master – June, 2008
Keywords	: Queueing Theory, Queueing Models, Monte Carlo Method

ABSTRACT

AN APPROACH TO QUEUEING THEORY VIA MONTE CARLO METHOD AND AN APPLICATION ON AUTOMOTIVE FIELD

The main frame of economics became increasing of population and meeting needs of human nowadays. During the process which assumes supplying high quality level of goods by manufacturer and consuming this high quality goods by human gather them together. Attitude of human such as joining this queues, changing mind after joining this queue, and never joining this queue is the main frame of this thesis.

Experiments can be operated via Monte Carlo method and Random deviated uniform numbers on queueing models like other kind of operations research subjects. This may be an obstacle for wasting time and extra money for experimenst.

İÇİNDEKİLER

Sayfa No.

TABLO LİSTESİ	III
ŞEKİL LİSTESİ	IV
KISALTMALAR	VI
SİMGELER	VII

1.GİRİŞ	1
---------------	---

2. KUYRUK TEORİSİ

2.1. Kuyruk Teorisinin Tarihçesi	3
2.2. Kuyruk Teorisinin Temel Kavramları	3
2.3. Kuyruk Teorisini Oluşturan Süreçler	6
2.3.1. Geliş Süreci	6
2.3.2. Kuyruk Konfigürasyonu	7
2.3.3. Kuyruk Disiplini	7
2.3.4. Servis Disiplini	8
2.3.5. Servis Olanağı	9
2.4. Kendal – Lee – Taha Notasyonu	11
2.5. Denge Durumu (Steady State) – Geçici Durum (Transient State)	13
2.6. Kuyruk Modellerinde Poisson Süreci Ve Üssel Dağılım Özelliği	14
2.7. Kuyruk Modelleri Ve Karşılaştırılmaları	17
2.7.1. Üssel Servis Zamanlı Modeller	17
2.7.1.1. M/M/1 Modeli İçin Sistem Performans Ölçüleri	18
1.7.1.1.1. Little Formülü	28
2.7.1.2. M/M/c Modeli İçin Sistem Performans Ölçüleri	29
2.7.1.3. M/M/c/K Modeli İçin Sistem Performans Ölçüleri	30
2.7.1.4. M/M/1/K Modeli İçin Sistem Performans Ölçüleri	32
2.7.1.5. M/M/c/c Modeli İçin Sistem Performans Ölçüleri	33
2.7.1.6. M/M/∞ Modeli İçin Sistem Performans Ölçüleri	35
2.7.1.7. PASTA Teoremi	36
2.7.2. Earlang Servis Zamanlı Kuyruk Modeli	36
2.7.2.1. M/E _k /1 Modeli İçin Sistem Performans Ölçüleri	36
2.7.3. Deterministik Servis Zamanlı Kuyruk Modeli	38
2.7.3.1. M/D/1 Modeli İçin Sistem Performans Ölçüleri	38
2.7.4. Genel Servis Zamanlı Kuyruk Modeli	38
2.7.4.1. M/G/1 Modeli İçin Sistem Performans Ölçüleri	38
2.7.5. Yığınsal Kuyruk Modelleri	39
2.8. Doğum – Ölüm Süreci	40
2.9. Öncelikli Servis Disiplini	45
2.9.1. Çok Kanallı Tam Öncelikli Olmayan Modeller İçin Sistem Performans Ölçüleri	45
2.9.2. Tek Kanallı Tam Öncelikli Olmayan Modeller İçin Sistem Performans Ölçüleri	46
2.9.3. Tam Öncelikli Modeller İçin Sistem Performans Ölçüleri	46

3. SİMÜLASYON YÖNTEMİ	48
3.1. Simülasyonun Tanımı ve Yöntemin Tarihçesi	48
3.2. Simülasyonun Yöneylem Araştırmasındaki Yeri Ve Önemi	49
3.3. Simülasyon Tekniğinin Kullanıldığı Alanlar	52
3.4. Simülasyonla İlgili Tanımlar	55
3.5 Simülasyon Modelleri	57
3.5.1. Statik Simülasyon Modeli	57
3.5.2. Deterministik Simülasyon Modeli	57
3.5.3. Stokastik Simülasyon Modeli	58
3.5.4. Dinamik Simülasyon Modeli	58
3.6. Simülasyon Modellerinde Kullanılan Gözlemler	58
3.7. Simülasyonun Kullanım Amaçları	59
3.8. Simülasyon Sürecinde İzlenmesi Gereken Adımlar	61
3.8.1. Problemin Formülasyonu Ve Çalışmanın Planlanması	63
3.8.2. Verilerin Toplanması Ve Modelin Tanımlanması	63
3.8.3. Modelin Geçerliliğinin Doğrulanması	64
3.8.4. Modelin Bilgisayarda Programlanması Ve Doğrulanması	64
3.8.5. Test Çalışmasının Yapılması	65
3.8.6. Modelin Geçerliliğinin Doğrulanması	66
3.8.7. Denemelerin Planlanması	66
3.8.8. Denemelerin Yapılması Ve Modelin Çalıştırılması	66
3.8.9. Sonuçların Analizi Ve Yorumu	67
3.8.10 Çıktıların Dökümantasyonu	67
3.9. Simülasyon Yönteminin Avantajları Ve Dezavantajları	67
3.10. Simülasyon Türleri	70
3.10.1. Monte Carlo Simülasyonu	70
3.10.2. Sistem Simülasyon Modelleri	71
3.10.3. İlkesiz Simülasyon Modeli	71
3.10.4. Oyun Modelleri	71
3.11. Simülasyon Dilleri	71
4. MONTE CARLO SİMÜLASYON METODU	74
4.1. Monte Carlo Simülasyon Metodunun Tarihçesi	74
4.2. Monte Carlo Simülasyon Metodunun Tanımı	75
4.3. Rassal Sayıların Üretilmesi Ve Test Edilmesi	76
4.4. Rassal Sayı Üretme Yöntemleri	78
4.4.1 Uygunluk Yöntemi	78
4.4.2. Ters Dönüşüm Yöntemi	80
4.4.2.1. Ters Dönüşüm Yöntemini Kullanarak Bilinen Olasılık Dağılımlarından Rassal Değişkenlerin Üretilmesi	82
4.4.2.1.1. Üssel Dağılım	82
4.4.2.1.2. Earlang Dağılımı	83
4.4.2.1.3. Uniform Dağılım	83
4.4.2.1.4. Weibull Dağılımı	84
4.4.2.1.5. Cauchy Dağılımı	85

4.4.2.1.6. Geometrik Dağılım	85
4.4.2.1.7. Pareto Dağılımı	86
4.4.2.1.8. Normal Dağılım	86
4.4.2.1.9. X^2 Dağılımı	87
4.4.3. Kabul – Red Yöntemi	88
4.4.3.1. Kabul – Red Yöntemi Yardımıyla Bilinen Olasılık Dağılımlarından Rassal Değişken Üretilmesi	89
4.4.3.1.1. Poisson Dağılımı	89
4.4.3.1.2. Gamma Dağılımı	90
4.4.3.1.3. Üçgen Dağılım	92
4.4.4. Konvolüsyon Yöntemi	93
4.4.4.1. Konvolüsyon Yöntemi Yardımıyla Earlang Dağılımından Rassal Değişkenlerin Türetilmesi	93
4.4.5. Ayırma Yöntemi	94
4.5. Örnek Hacminin (Deneme Sayısının) Belirlenmesi	95
4.5.1. Ortalama İle Deneme Sayısının Bulunması	96
4.5.2. Standart Sapma İle Deneme Sayısının Bulunması	96
4.6. Rassal Sayı Üreteçlerinin Testi	98
4.6.1. X^2 Testi	99
4.6.2. Kolmogorov – Smirnov Testi	100
4.6.3. Çiftler Testi	101

5. TEK KANALLI KUYRUK MODELİNDE MONTE CARLO

SİMÜLASYON METODUNUN BİR İŞLETMEDE UYGULANMASI 102

5.1. Uygulamanın Yapıldığı İşyeri İle İlgili Bilgiler	102
5.2. Uygulamanın Amacı	102
5.3. Araştırmada Kullanılan Yöntem	103
5.4. Sistemin Tanımlanması	103
5.5. Verilerin Kullanılması	104
5.6. Gelişler Arası Sürenin Poisson Dağılımına Uygunluğunun Testi	111
5.7. Servis Sürelerinin Üssel Dağılıma Uygunluğunun Testi	115
5.8. Birikimli Olasılık Fonksiyonlarının Oluşturulması Ve Monte Carlo Simülasyon Metodunun Uygulanması	117
5.9. Örnek (Deneme) Sayısının Belirlenmesi	119
5.10. Monte Carlo Simülasyon Metodunun Uygulanması	121

6. SONUÇLAR..... 126

EKLER 130

KAYNAKÇA 132

TABLO LİSTESİ

	Sayfa No:
Tablo 1 : Kuyruk Modellerine Bazı Örnekler	4
Tablo 2 : Kuyruk Notasyonu	12
Tablo 3 : t Anında Sistemdeki Müşteri Sayısı ve Olayların Olma Olasılıkları	20
Tablo 4 : Doğum – Ölüm Sürecinde Denge Durumu Denklemleri	44
Tablo 5 : Denge Durumu Denklemlerinden Olasılık Değerlerinin Elde Edilmesi	44
Tablo 6 : Yöneylem Araştırmasında Kullanılan Yöntemlerin Üstünlükleriyle İlgili Yapılmış Bir Araştırma	50
Tablo 7 : Simülasyonun Kullanıldığı Alanlar	54
Tablo 8 : χ^2 Testi Örneği	99
Tablo 9 : Müşterilerin Gelişler Arası Süresi, Sistemde Geçirdikleri Ve Servise Başladıkları Süreler	106
Tablo 10 : Müşteriler İçin Servis Süresi, Servisin Bitişi Ve Sunucunun Boş Kaldığı Süreler	107
Tablo 11 : Müşterilerin Sistemde Geçirdiği Süre, Bekleme Süresi, Kuyruktaki Ve Sistemdeki Müşteri Sayıları	108
Tablo 12 : χ^2 Testinde Kullanılacak Gözlenen Ve Beklenen Frekans Değerleri	113
Tablo 13 : Gelişler Arası Sürelerin Poisson Dağılımına Uygunluğunun Testi Sonuçları	114

Tablo 14 :	Servis Sürelerinin Üssel Dağılıma Uygunluğunun Testi Sonuçları	115
Tablo 15 :	Gelişler Arası Süreler İçin Rassal Sayı Aralıklarının Oluşturulması	118
Tablo 16 :	Servis Süreleri İçin Rassal Sayı Aralıklarının Oluşturulması	119
Tablo 17 :	Monte Carlo Deneyinde Rassal Sayıların Kullanılması	121
Tablo 18 :	Monte Carlo Deneyi Sonucunda Elde Edilen Veriler (1)	123
Tablo 19 :	Monte Carlo Deneyi Sonucunda Elde Edilen Veriler (2)	125

ŞEKİLLER LİSTESİ

Sayfa No :

Şekil 1	: Maliyet – Hizmet Düzeyi	5
Şekil 2	: Servis Konfigürasyonları	10
Şekil 3	: Denge Durumu – Geçici Durum	14
Şekil 4	: Doğum – Ölüm Süreci	41
Şekil 5	: Alternatif Problem Çözüm Yöntemleri	51
Şekil 6	: Simülasyon Modellerinin Sınıflandırılması	57
Şekil 7	: Simülasyon Çalışmasında İzlenen Adımlar	62
Şekil 8	: Ters Dönüşüm Yöntemi	81
Şekil 9	: Ayırma Yöntemi	94
Şekil 10	: Stokastik Limit	98
Şekil 11	: Kuyruk Sisteminin Şematik Olarak Gösterimi	104
Şekil 12	: Müşterilerin Sisteme Gelişler Arası Süresini Gösteren Grafik	109
Şekil 13	: Servis Sürelerini Gösteren Grafik	110
Şekil 14	: Müşterilerin Sistemde Geçirdiği Süreleri Gösteren Grafik	111
Şekil 15	: Monte Carlo Deneyi Sonucunda Gelişler Arası Süreyi Gösteren Grafik	122
Şekil 16	: Monte Carlo Deneyi Sonucunda Servis Sürelerini Gösteren Grafik	124

KISALTMALAR

FIFO	First In First Out
FCFS	First Come First Served
LCFS	Last Come First Served
SIRO	Service In Random Order
GD	General Discipline
PR	Priority Discipline
PASTA	Poisson Arrivals See Time Average
PSPO	Preemptive Service Priority Order
NPPS	Tam Öncelikli Servis Disiplini
ORSA	Operations Research Society Of America
RAND	The Research And Development Corporation
MS	Microsoft
RS	Rassal Sayı
GAS	Gelişler Arası Süre
S. Başl. A.	Servisin Başladığı An
S. Bit. A.	Servisin Bittiği An
SBKS	Sunucunun Boş Kaldığı Süreler
MSGs	Müşterilerin Sistemde Geçirdiği Süre
MKGs	Müşterilerin Kuyrukta Geçirdiği Süre
KMS	Kuyruktaki Müşteri Sayısı
SMS	Sistemdeki Müşteri Sayısı

SİMGELER

λ	Sisteme Ortalama Varış Oranı
λ_{eff}	Sisteme Efektif Varış Oranı
μ	Ortalama Servis Süresi
ρ	Geliş Hızı
M	Markovian – Memoryless Property
E_k	Earlang Dağılımı
H_k	Hiperüssel Dağılım
G₁	Gelişler Arası Sürenin Bağımsız Dağılımı
G	Servis Sürelerinin Genel Dağılımı
L	Sistemdeki Ortalama Müşteri Sayısı
L_q	Kuyruktaki Ortalama Müşteri Sayısı
W	Sistemde Geçen Ortalama Süre
W_q	Kuyrukta Geçen Ortalama Süre
P₀	Sistemin Atıl Kalma Olasılığı
P_n	Sistemde N Sayıda Müşteri Olma Olasılığı
W_k	K Sınıfına Ait Tam Öncelikli Model İçin Ortalama Sistemde Bekleme Süresi
L_k	K Sınıfına Ait Tam Öncelikli Model İçin Sistemde Beklenen Müşteri Sayısı

GİRİŞ

İnsanın, ihtiyaçlarını karşılamak için günlük yaşantısında bazı toplumsal kurallara uyması gerekmektedir. Bu doğrultuda hepimiz çoğu zaman pek çok sistemle ilişki içindeyizdir.

İlişki içinde olduğumuz bu sistemlerde bizim gibi pek çok sayıda insanla da aynı amaç doğrultusunda bir araya geliriz. Bu durumda mal veya hizmet ihtiyacımızın anında karşılanamaması, mevcut hizmet sunucusunun her bir müşteri için belli bir zaman ayırması ve geçen zaman diliminde sisteme yeni müşterilerin katılması ile sıra veya kuyruk adını verdiğimiz yeni bir sistem oluşur.

Her birimizin toplumsal kural gereği; farkında olmadan katıldığımız bu sistemin adı Kuyruk Teorisidir. Diğer bir deyişle Bekleme Hattı Modelleri, hem bu toplumsal düzenin sağlanmasına katkıda bulunmakta hem de müşteriler için beklemekten doğacak maliyeti ve işletmeler için ek hizmet kanalları açmak veya müşteri memnuniyeti sağlamak için doğacak olan maliyeti minimize etmeyi amaçlamaktadır.

Temelleri geçtiğimiz yüzyılın başında atılan teori işletmeler tarafından sıklıkla kullanılmakta pek çok alanda uygulama şansı bulmaktadır.

Yirminci yüzyılın başlarında bir bilim adamının telefon görüşmeleri üzerinde yaptığı araştırma günümüzde uçakların hava alanlarına iniş ve kalkışlarıyla, otoyolda ücret ödemek için bekleyen otomobillerle, süpermarket kasalarında ödeme yapmak için bekleyen müşterilerle açıklanıyor.

Benzer şekilde bir bilim adamı olan Ulam'ın hastalıktan kurtulma evresinde evinde vakit geçirirken oynadığı soliter oyunu, onu ileride nükleer silahların ve uzay araştırmacılığının önünü açacak bir yöntemin keşfine itti.

Gerçek denemeler yapmak yerine üretilen rassal sayılar üzerinde, bazı varsayımların yerine getirilmesi koşuluyla çalışma yapmanın daha az maliyet ve daha az

zaman harcaması kısa bir süre sonra bu buluşun hızlı bir şekilde yayılmasına ve yöneylem araştırması konuları içinde daha fazla önem taşımasına sebep oldu.

Araştırmanın birinci bölümünde Kuyruk Teorisi açıklanmıştır. Teorinin tarihi süreç içerisindeki gelişiminden bahsedilmiş; teorik olarak karşılaşılabileceğimiz modeller, modellerin dayandığı varsayımlar ve bu varsayımları destekleyen özellikler açıklanmıştır.

Araştırmada kullanılan bir diğer teknik de Monte Carlo Simülasyon Tekniğidir. Araştırmanın ikinci bölümünde simülasyon tekniğinin genel özellikleri üzerinde durularak simülasyonun kullanım alanları, avantaj ve dezavantajları, simülasyon modellerinin süreçleri anlatılmıştır.

Araştırmanın dördüncü bölümünde simülasyon yöntemine ek olarak Monte Carlo tekniğinin savaş yıllarında nasıl ortaya çıktığı ve nasıl geliştiği, rassal sayı üreticilerine neden ihtiyaç duyulduğu, rassal sayı üretme yöntemleri ve üretilen rassal sayıların test edilmesi konuları açıklanmıştır.

Araştırmanın son bölümündeyseniz Monte Carlo Simülasyon Metodu yaklaşımıyla Kuyruk Teorisi Modellerinden uygun olanı incelenmiş ve bir işletme uygulaması yapılarak araştırmaya son verilmiştir.

2. KUYRUK TEORİSİ

2.1. Kuyruk Teorisinin Tarihçesi

Kuyruk Modelleri ile ilgili ilk çalışma Danimarkalı bir elektrik mühendisi olan A. Karl Earlang tarafından yapılmasına rağmen bu konuda yayınlanan ilk eser Johansen'in 1907'de yazdığı 'Bekleme Zamanı ve Telefon Etme Sayısı' (Waiting Times And Number Of Calls) başlığını taşıyan makalesidir¹. Earlang çalışmasını telefonu kullananların sayısı, telefonla konuşmaların süresi ve telefonun meşgul kaldığı süre üzerine yoğunlaştırmıştır. Araştırmasının ilerleyen zamanlarında Telefonu kullanmak üzere telefon başına varan müşteri sayısının Poisson, konuşma sürelerinin ise Üssel Dağılıma uyduğunu ilk kez fark edip çalışmasını bu alanda yoğunlaştırmıştır. Bu sebeple teorisinin kurucusunun Earlang olduğunu söyleyebiliriz.

Earlang'ın bu çalışması pek çok bilim adamına ilham kaynağı olmuş ve bu konuda yapılan çalışmalar hız kazanmıştır. 1927 yılında C. Molina'nın "Application Of The Theory Of Probability To Telephone Trunking Problems" ve 1928 yılında C. Fry'n "Probability And Its Engineering Uses" adlı çalışmaları da Earlang'ın telefon sistemleriyle ilgili yaptığı çalışmaların tamamlayıcı makaleleri niteliğindedir.

1930 ve 1950 arası teoriye önemli katkılarda bulunuldu. Bu dönemde Crommelin telefon sisteminde bekletilen telefonlarla ilgili olasılık formülleri geliştirdi. Pollaczek ve Khintchine Poisson varış süreli, değişen ve sabit zaman servis süreli ve tek kanallı sıra bekleme sistemleri için Pollaczek-Khintchine formülünü geliştirdiler².

2.2. Kuyruk Teorisinin Temel Kavramları

Kuyruk teorisinden ve modellerinden söz edebilmek için öncelikle bazı tanımlamaların yapılması gerekmektedir.

¹ Halil Sarıaslan, *Sıra Bekleme Sistemlerinde Simülasyon Tekniği*, Ankara, Ankara Üniversitesi Siyasal Bilgiler Fakültesi Yayınları, 1986, s:6

² Sarıaslan, s:6

Kuyruk sisteminin oluşabilmesi için müşteri (costumer) ve hizmet sağlayıcı - sunucu (server) tanımının ve özelliklerinin bilinmesinde fayda vardır. Müşteri, en genel tanımıyla belli bir mal veya hizmetten faydalanmak isteyen ünedir. Genelde insan olarak algılansa da bu süper markette kuyruğa giren bir müşteri, onarım bekleyen bir makine veya daktilo edilmeyi bekleyen bir metin de olabilir. Sunucu veya hizmet sağlayıcılar için de aynı özellik geçerlidir. Süper markette ödemeleri alan bir kasiyer, mekanik onarımı yapan bir işçi veya bir makine olabileceği gibi sayfaları yazmak için görevlendirilmiş bir sekreter servis sağlayıcı olarak düşünülebilir.

Tablo1
Kuyruk Modellerine Bazı Örnekler

Müşteri	Sunucu	Kuyruk	Mümkün Durumlar
Otomobil	Ödeme noktasında ödeme alan kasiyer veya otomatik geçiş sistemi.	Ödeme yapmak için bekleyen otomobiller	Ödeme yapmak için bekleme, servis sunum maliyeti
Bozuk makineler	Tamirci	Onarım bekleyen makineler	Onarım süresi, bozuk makinelerin beklemesinden doğan maliyet
Hastalar	Doktorlar	Tedavi bekleyen hastalar	Tedavi maliyeti, bekleyen hastaların sağlık durumları.
Uçaklar	Kalkış-iniş pistleri	İniş ve kalkış için bekleyen uçaklar	Yakıt maliyeti, müşteri memnuniyeti.
Telefon aramaları	Telefon hattı	Meşgul sinyali alan müşteriler.	İhtiyaç duyulan fazla hat sayısı, müşteri şikayetleri.

Kaynak: Principles of Operations Research for Management, Frank S. Budnick, Richard Mojena,1977, s:433

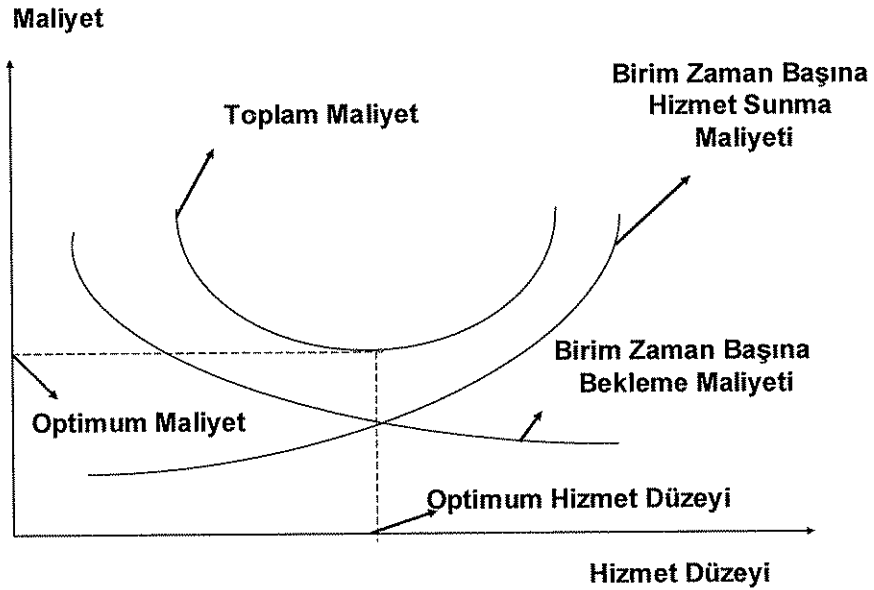
Mal ve hizmetten faydalanmak için servis noktasına gelen müşterilerin bu isteklerinin anında karşılanamaması ve her bir servis noktasında müşterilere belli bir vakit ayrılması kuyruğun oluşmasına sebep olacaktır.

Kuyruk modellerinin analizindeki amaçlar şunlardır³ :

³ Osman Halaç, **Kantitatif Karar Verme Teknikleri**, 4. Basım, İstanbul, İstanbul Üniversitesi, 1983, s: 294

- 1) Çalışma kuralları ve karakteristik parametreler dizisinin belirlenmesi.
- 2) Zaman kayıplarının ve gecikme koşullarının belirlenmesi.
- 3) Gerekli servis ve sürecin çalıştırılmasından sağlanan gelir ve maliyet arasında optimal dengenin bulunması.
- 4) Çözümün uygulanabilirlik derecesi, incelenen sistemin politika seçeneklerine bağlıdır.

Kuyruk Modellerinde karar sürecinin ana unsurları beklemekten doğacak müşteri memnuniyetsizliğini ve daha kısa sürede hizmet etmek için ek servis kanalları açmanın yaratacağı maliyeti minimize etmek yönünde olmaktadır. Beklemeyi tamamıyla ortadan kaldıramayız. Gerçekte başarabileceğimiz tek şey beklemenin ters etkisini kabul edilebilir düzeye indirmektir⁴.



Şekil 1. Maliyet – Hizmet Düzeyi

⁴ Hamdy A. Taha, **Yöneylem Araştırması**, 6. Basım, Fayetteville – ABD, Arkansas Üniversitesi, 2000, s:597

2.3. Kuyruk Teorisini Oluşturan Süreçler

Kuyruk problemlerinin çözülmesi için bazı süreçlerin tanımlanması gerekmektedir. Bu süreçleri şu şekilde sıralayabiliriz;

2.3.1. Geliş Süreci

Bu süreç, müşterilerin kuyruk sistemine gelişini ifade eder. Bu süreç sekiz farklı özelliklerle ifade edilebilir⁵.

- 1) Müşterilerin geldiği kaynak bir ya da birden fazla olabilir
- 2) Müşterilerin geldiği kaynak sonlu veya sonsuz olabilir.
- 3) Kuyruk sistemine gelişler tek veya yığınsal olarak gerçekleşebilir.
- 4) Gelişler sistem tarafından tamamen kontrol edilebildiği, kısmen kontrol edilebildiği veya hiçbir şekilde kontrol edilemediği söylenebilir.(Üretim bandında birleştirilmeyi bekleyen parçalar tamamen kontrol edilebilir, uçakların havaalanına gelişi kısmen kontrol edilebilir, otomobillerin otobanlardaki ödeme noktasına varışı kontrol edilemeyen süreçlere örnektir.)
- 5) Müşterilerin sisteme varışı deterministik veya stokastik karakterlidir.
- 6) Müşterilerin stokastik olarak sisteme gelişi ampirik veya teorik bir olasılık fonksiyonuyla tanımlanır.(Varış sürecini tanımlamak için en çok Poisson dağılımı kullanılmaktadır. Çünkü çoğu örnekte müşterilerin sisteme varış sürecinin Poisson dağılımına uyduğu görülür.)
- 7) Geliş süreci bağımlı ve bağımsız olarak tanımlanabilir. Duruma bağlı olarak haftanın belli günlerinde veya günün belli saatlerinde trafik ışıklarında oluşan kuyruklar buna örnek olarak verilebilir.

⁵ Frank S. Budnick, Richard Mojena ve Thomas E. Vollman, **Principles Of Operations Research For Management**, 18. Basım, Homewood Ill, R. D. Irwin, 1977, s:434

8) Geliş süreci durağan olabilir veya zaman içinde değişir. Süreç durağansa parametreler ve olasılık yoğunluk fonksiyonunun formu sabittir ve tahmin edilebilir. Genel varsayım bu yöndedir; aksi durumda bu matematiksel büyüklüklerin tahmini zorlaşacaktır.

2.3.2. Kuyruk Konfigürasyonu

Kuyruk konfigürasyonu, sistemde oluşan kuyruk sayısını ve bu kuyrukların servis sağlayıcısıyla olan ilişkisini gösterir. Servis sağlayıcısının hizmet sunduğu noktaya servis kanalı denir. Tek bir kuyruktan oluşan sistemde bir servis kanalı olabileceği gibi tek bir kuyruktan oluşan sistemde birden fazla servis kanalı da bulunabilir. Birden fazla servis kanalının bulunduğu sistemde de genel olarak aynı sayıda kuyruğun oluşması beklenir. Kuyruklar fiziksel olarak bir tek yerde bulunabileceği gibi birkaç değişik yerde veya havaalanı ve otel rezervasyonlarında olduğu gibi sadece liste şeklinde ve kağıt üzerinde de olabilir⁶.

2.3.3 Kuyruk Disiplini

Daha önce de belirtildiği gibi varış süreci hizmet kapasitesinden büyükse sistemde kuyruk oluşur. Kuyruk disiplini de gelen müşterilerin bir kuyruğu seçme veya seçmeme davranışı ve kuyrukta bekleme süresi içinde göstereceği davranışı kapsar. Kuyruk disiplini aşağıdaki şekilde tanımlanır;

1) Sistemin kapasitesi doluysa kuyruğa yeni katılan birim geri çevrilir (Rejected). Telefon görüşmesinde yapılan aramada meşgul sinyalinin alınması veya tüm park yerleri doluyken park etmek için gelen bir sürücünün geri çevrilmesi gibi durumlar buna örnektir. (Uygulamada sistemin burada sona ermediği bilinir. Çünkü aramayı yapan kişi meşgul sinyalini duyduktan sonra telefonu kapatıp veya bir süre bekleyip tekrar arama yapacaktır. Aracını park edemeyen sürücü de bir süre bekleyip aynı kuyruğa baştan tekrar katılacaktır.)

⁶ Budnick, Mojena ve Vollman, s:434

2) Hizmet almak için gelen müşterilerin ortalama bekleme süresi fazlaysa veya kuyruk normalden çok uzunsa müşteri isteksiz davranabilir veya kuyruğa hiç katılmadan geri dönebilir (Balk).

3) Eğer sisteme gelen müşteri kuyruğa katılmışsa ve buna değmeyeceğini düşünüyorsa zaman sınırlılığı sebebiyle başka bir zaman tekrar gelmek üzere kuyruğu terk edebilir (Reneged).

4) Çok kanallı kuyruk sistemlerinde sabırsız müşteriler kısa olduğunu düşündükleri kuyruklara katılabilirler ve bu kuyruklar içinde sürekli yer değiştirebilirler (Jockey).

5) Çok kanallı kuyruk sistemlerinde müşterilerin genel karar kriteri en kısa olduğunu düşündükleri kuyruğa katılmak veya rassal olarak herhangi bir kuyruğa katılmayı göze almak şeklinde olur⁷.

2.3.4 Servis Disiplini

Servis disiplini kuyrukta bekleyen müşterilerin hizmete dahil edilme sırası veya önceliğidir. Genel olarak beş şekilde gerçekleşir.

1) FIFO (First In First Out) veya FCFS (First Come First Served) olarak adlandırılan ilk gelen müşteriye önce hizmet edilir disiplinidir.

2) LCFS (Last Come First Served) olarak adlandırılan son gelen müşteriye ilk hizmet edilir şeklindedir. Örneğin asansöre en son binen müşteri gideceği katta ilk inen olacaktır veya arabalı vapura en son binen sürücü gideceği limanda ilk inen olacaktır.

3) SIRO (Service In Random Order) rassal olarak seçilen herhangi bir müşteriye hizmet etme sırasıdır. Çekiliş yöntemiyle önceliğin belirlenmesi buna örnek olarak verilebilir.

4) Hizmet almak için bekleyen müşterilerin yarışmayla (Round Robin) öncelik sıralarının belirlenmesi.

⁷ Budnick, Mojena ve Vollman, s:435

5) Öncelikli servis disiplini (Priority Service) iki şekilde gerçekleşir. İlki sisteme önceliği olan yeni bir birim katıldığı zaman hizmet gören müşterinin hizmeti kesilir ve yüksek öncelikli olan birime hizmet edilir (Preemptive Priority Discipline). Daha sonra hizmeti kesilen birim tekrar kaldığı yerden hizmet almaya devam eder. İkincisi, hizmet gören müşterinin hizmetinin bitmesi beklenir (Non - Preemptive Priority Discipline) ve akabinde yüksek önceliği olan müşteriye hizmet edilir. Hastanelerdeki acil servisler hastaların durumuna göre öncelikli servis disiplini uygularlar⁸.

2.3.5. Servis Olanakları

Servis olanakları aşağıdaki şekilde olabilir⁹.

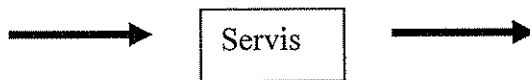
1) Sıfır, bir veya çok kanallı servis olanakları. Süper marketler örnek olarak verilebilir. Müşteriler istedikleri ürünü raftan kendileri self servis yöntemiyle sıfır kanallı servis olanaklarıyla alırlar ve ödemeyi kasalarda çok kanallı servis olanaklarıyla yaparlar.

2) Servis kanalları paralel, seri veya her ikisini içerecek şekilde olabilir.

3) Geliş sürecinde olduğu gibi servis süresi de deterministik veya olasılıksal olabilir. Otomasyona bağlı makinelerle yapılan üretimin deterministik olduğuna daha önce değinmiştik. Uygulamada çoğu servis süresinin teorik veya ampirik olarak bir olasılık dağılım fonksiyonundan geldiği bilinmektedir. Üssel dağılım servis sürelerinin tanımlanmasında en çok karşılaştığımız olasılık fonksiyonudur.

4) Servis kanalında hizmet sunan makinenin bozulması gibi hizmet sunan insanın da hastalanabileceği düşünülmelidir.

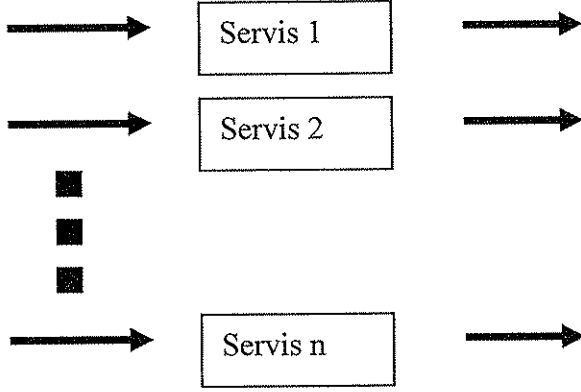
a) Tek Kuyruk – Tek Kanal



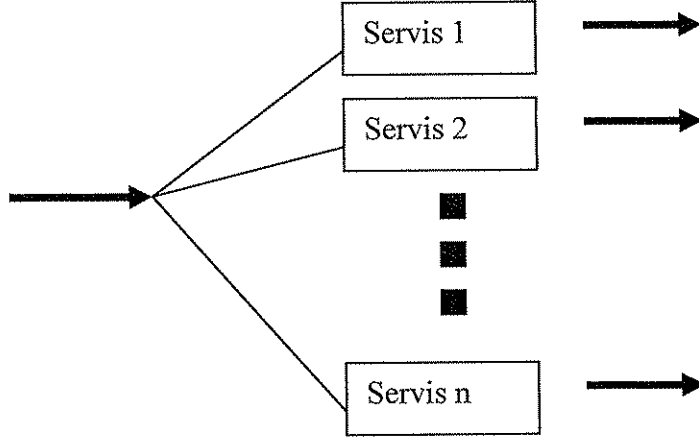
⁸ Budnick, Mojena ve Vollman, s:435

⁹ Budnick, Mojena ve Vollman, s:436

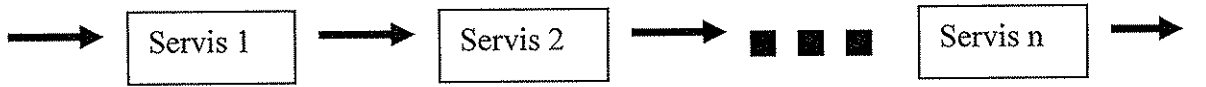
b) Çoklu Kuyruk – Çoklu Kanal



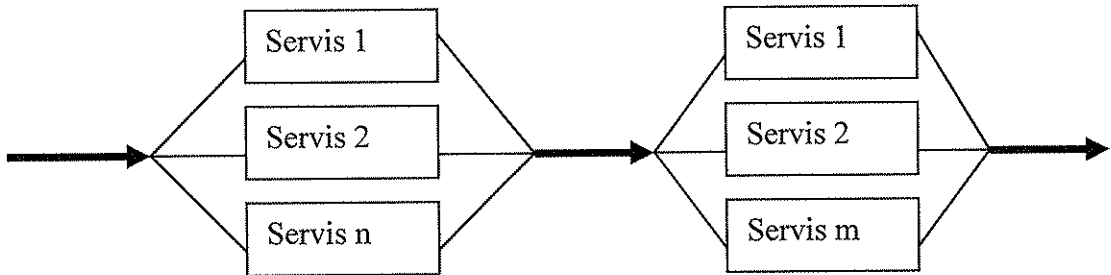
c) Tek Kuyruk – Çoklu Kanal



d) Tek Kuyruk - Seri Kanal



e) Seri ve Paralel Halde Çoklu Kuyruk



Şekil 2. Servis Konfigürasyonları

Yukarıda tanımlaması yapılan teorinin temel öğelerine bazı eklentilerin yapılması gerekmektedir. Bu durum, problemlerin çözümünde çoğu zaman belirtilmemekte fakat sabit olduğu ve her problemde koşul olarak yerine getirildiği varsayılmaktadır.

Bu varsayımları şu şekilde sıralayabiliriz:

1) Varış oranı λ 'nın servis oranı μ 'den küçük olması gerekmektedir. Aksi takdirde varış oranı veya sayısı büyük bir hızla artarken servis oranı bu artışı karşılayamayacak ve kuyruk sınırlanamayan bir şekilde büyüyecektir.

2) Bir ünitenin servis alıp sunucudan ayrılmasından sonra yerine kuyruktan başka bir ünitenin gelmesine kadar geçen sürenin 0'a çok yakın bir zaman aralığı olduğu varsayılmakta ve bu süre dışlanmaktadır¹⁰.

3) Çok kanallı kuyruk modellerinde hizmet sağlayan sunucuların hizmet sürelerinin birbirine eşit olduğu varsayılır¹¹.

2.4. Kendall – Lee – Taha Notasyonu

D. Kendall, çok kanallı kuyruk modellerinin daha kolay tanımlanabilmesi için 1953 yılında genel bir notasyon önermiştir. Bu notasyon gelişler arası sürenin dağılımını, servis süresinin dağılımını ve kanal sayısının dağılımını tanımlamaktadır. 1966 yılında A. Lee bu notasyona dördüncü ve beşinci elemanları olan servis disiplini ve sistemde bulunabilecek maksimum ünite sayısını eklemiştir. 1968 yılında notasyon H. Taha'nın altıncı eleman olan geliş kaynağının büyüklüğünü eklemesiyle en genel halini almıştır¹².

¹⁰ I. Mitrani, **Modelling Of Computer And Communication Systems**, 8. Basım, Cambridge – İngiltere, Cambridge University Press, 1987, s: 70

¹¹ Taha, s: 618

¹² Taha, s: 619

Bu notasyon řu řekilde gsterilebilir¹³:

(a/b/c) : (d/e/f)

- a : Geliřlerarası srenin daęılımı
- b : Servis sresi daęılımı
- c : Sistemdeki kanal sayısı
- d : Servis disiplini
- e : Sistemde bulunabilecek maksimum nite sayısı
- f : Geliř kaynaęının byklę

Tablo 2
Kuyruk Notasyonu

a (Geliřlerarası srenin daęılımı) b (Servis sresinin daęılımı)	M	Poisson Daęılımı
	M	ssel Daęılım
	E _k	Earlang Daęılımı
	H _k	Hiperssel Daęılım
	GI	Geliřlerarası Srenin Baęımsız Daęılımı
	G	Servis Srelerinin Genel Daęılımı
d (Servis Disiplini)	FIFO	İlk Gelen İlk Hizmet Grr
	LCFS	Son Gelen İlk Hizmet Grr
	SIRO	Servis Sırası Rassal Seilir
	GD	Genel Disiplin
	PR	ncelikli Servis Disiplini

Kaynak: Fundamentals of Queueing Theory, Donald Gross – Carl M. Haris, 1998, s:8

¹³ Taha, s: 618

Örneğin (M/M/1) : (FIFO/∞/∞) olarak tanımlanan kuyruk modeli, gelişler arası sürenin dağılımının Poisson ve servis süresinin üssel olduğunu, sistemde bir kanal bulunduğunu, ilk gelen ilk hizmet görür prensibiyle hizmet verildiğini, sistemin kapasitesi ve müşterilerin geliş kaynağıyla ilgili bir sınırlandırmanın yapılmadığı gösterir.

Uygulamalarda çoğu zaman sistemin kapasitesiyle ve müşteri kaynağının geliş kaynağıyla ilgili sınırlandırma yapılmamışsa ve ilk gelen ilk servis görür disiplini geçerliyse sadece ilk üç simgenin kullanılması yeterli olacaktır. Yukarıdaki örnek için (M/M/1) notasyonu kuyruk modelinin tanımlanması için yeterlidir. Eğer sistemde hizmet verilecek müşteri sayısı ile ilgili bir sınırlandırma yapılmışsa bu gösterime kısıt sayısı da dahil edilebilir. Örneğin M/M/c/c şeklinde yapılan gösterim gelişler arası sürenin Poisson ve servis süresinin üssel dağılıma uyduğunu, sistemde c sayıda kanal olduğunu ve en fazla c sayıda müşteriye hizmet verilebileceğini gösterir. Bu gösterim kısa olmasından dolayı araştırmalarda bu şekilde kullanılır.

Ayrıca bu notasyonların dışında bazı gösterimlerin de atlandığını söyleyebiliriz. Örneğin grup halinde gelişlerin, seri haldeki kuyrukların ve duruma bağlı gelişlerin olduğu notasyon için bir genelleme yapılmamıştır. Bunun sebebi bu modeller için henüz sabit bir notasyon gösteriminin yapılamamış ve literatürde Kendall – Lee – Taha notasyonunun doğru kabul edilmiş olmasıdır¹⁴.

2.5. Denge Durumu (Steady State) - Geçici Durum (Transient State)

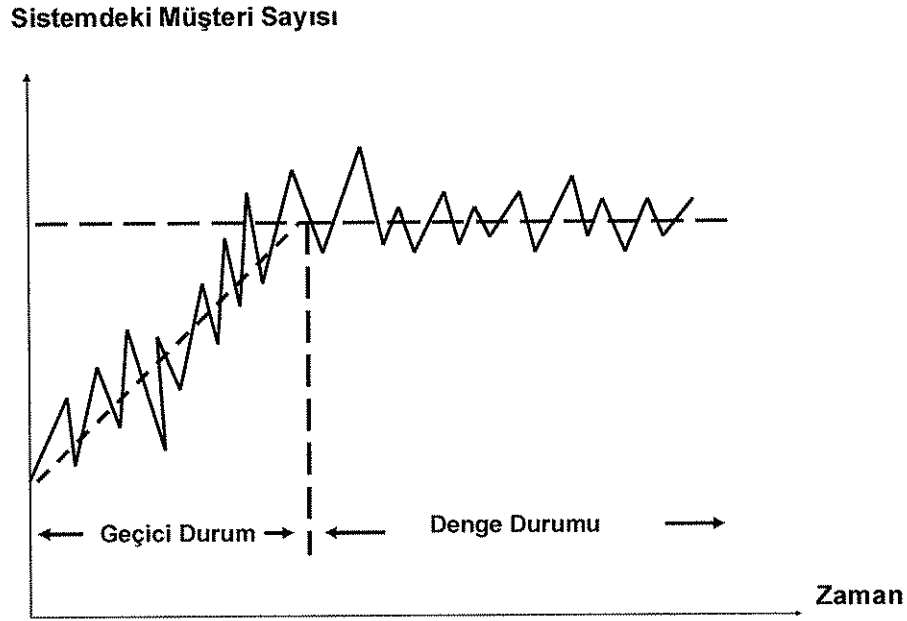
Kuyruk süreçlerinin analizinde durağan durumla geçici durum arasındaki farkın tanımlanması önem arz eder. Geçici durumda süreç zamana bağlıdır ve zaman içerisinde sistemin karakteristiklerinin tahmini gerçek veya gerçeğe yakın bir şekilde tahmin edilmesi güçleşir. Düzgün durumda ise sistemin karakteristikleri istatistiksel olarak dengededir. Diğer bir deyişle zamandan bağımsızdır. Geçici durumla genellikle kuyruk süreçlerinin başlangıç safhalarında karşılaşılır. Örneğin bir mağazanın açılışında veya indirim günlerinde kuyruğun başlangıç süreci farklı karakteristikler sergileyebilir.

¹⁴ Donald Gross, Carl M. Harris, **Fundamentals Of Queueing Theory**, 3. Basım, New York – ABD, A. Willey Publication, 1998, s: 8

Karşılaşılan durumlar genellikle hiç kuyruğun olmaması, kısa kuyruk oluşumu veya tam tersine uzun bir kuyruğun oluşumu şeklinde gözlemlenebilir. Bu durum sürecin ilk safhalarında parametrelerin tahminini etkileyecektir. Fakat belli bir süre geçtikten sonra sistemin genellikle durağan hale geldiği gözlemlenir¹⁵.

Bazı durumlarda kuyruk süreçleri zaman geçse dahi asla durağanlık göstermeyebilir. Bunun sebepleri:

- 1) Ortalama varış hızının ortalama servis süresinden fazla olması
- 2) Sürecin incelendiği zaman aralığının kısa olmasıdır¹⁶.



Şekil 3. Denge Durumu – Geçici Durum

2.6. Kuyruk Modellerinde Poisson Süreci Ve Üssel Dağılım Özelliği

Çoğu stokastik elemanlı kuyruk modelinde gelişler arası sürenin ve servis süresinin üssel dağılıma uyduğu kabul edilir; veya buna eşdeğer olarak geliş oranının ve servis oranının Poisson sürecine uygun olarak tanımlanabileceği kabul edilir. Poisson

¹⁵ Budnick, Mojena ve Vollman, s: 439

¹⁶ Budnick, Mojena ve Vollman, s: 440

dağılımının öncelikle gerekli varsayımları sağlaması gereklidir. Bu varsayımları aşağıdaki şekilde açıklayabiliriz;

1) Herhangi bir t anında bir müşterinin gelişi diğer müşterilerin gelişinden tamamen bağımsızdır.

2) Herhangi bir t anında müşterilerin gelişi zamandan bağımsızdır.

3) Herhangi bir t anında sisteme 1 tane müşteri gelir. (Sisteme t anında gelen müşteri sayısı ya 0'dır veya 1'dir. t anında 1'den fazla müşteri gelmediği varsayılır.)

$N(t)$, t anına bağlı olarak sistemde gerçekleşen toplam müşteri geliş sayısını göstermek üzere ($N(0) = 0$) $\{N(t), t \geq 0\}$ sürecinin aşağıdaki üç özelliği sağladığını varsayalım¹⁷:

1) t ve $t + \Delta t$ zaman aralığı içerisinde sisteme katılma olasılığı $\lambda \Delta t + o(\Delta t)$ olarak yazılır. Diğer bir gösterişle

$$P\{t \text{ ve } t + \Delta t \text{ zaman aralığı içinde sisteme geliş olayı}\} = \lambda \Delta t + o(\Delta t)$$

olarak yazılabilir. ($o(\Delta t)$ çok küçük bir zaman dilimi olduğundan dışlanması bir şeyi değiştirmeyecektir)

$$2) P\{t \text{ ve } t + \Delta t \text{ zaman diliminden birden fazla geliş olasılığı}\} = o(\Delta t) = 0$$

3) Gelişler ve gelişler arası süre istatistiksel olarak bağımsızdır, diğer bir deyişle süreç bağımsız artırımlara sahiptir.

Yukarıdaki koşulları sağlamak üzere bir Poisson dağılımına ait olasılık fonksiyonu:

¹⁷ Gross ve Haris, s: 17

$$P_n(t) = \frac{(\lambda t)^n e^{-\lambda t}}{n!}, n=0,1,2,\dots$$

şeklinde tanımlanır.

Belli bir kaynaktan sisteme katılan müşterileri a_j , sisteme ardışık olarak katılan müşteriler arasında geçen süreyi I_j , servis süresini de S_i sembolleriyle gösterelim. ($I_1 = a_1 - 0, I_2 = a_2 - a_1, \dots, I_j = a_j - a_{j-1}$). a_j değişkeninin rassal olduğu varsayımı altında sisteme katılan müşteri sayısı tam olarak bilinemesi de aradan geçen sürenin istatistiksel olarak tanımlanması yapılabilir¹⁸.

Ardışık varışlar arasında geçen sürenin dağılımı;

$$P\{I_j < \chi\} = 1 - e^{-\lambda\chi}, \quad j \geq 1$$

servis süresinin dağılımını da,

$$P\{S_i < \chi\} = 1 - e^{-\mu\chi}, \quad i \geq 1 \text{ şeklinde gösterebiliriz.}$$

Üssel dağılımın özelliğini anlatmak için kısa bir örnek verelim. Elimizde zaman ayarlı bir bomba olduğunu ve bu bombanın X zaman sonra otomatik olarak patlayacağını düşünelim. Aradan geçen X süresinin üssel dağılıma uyduğu kabul edilir. Dağılımı üssel dağılım notasyonu ile yazmamız gerekirse:

$$P\{X < \chi\} = 1 - e^{-\lambda\chi}, \quad 0 < \lambda < \infty$$

$t = 0$ anında zamanı başlattığımızı ve $t = X$ anında patlamanın gerçekleşeceğini varsayalım. Patlama henüz gerçekleşmeden, yani geri sayımın ilk dakikalarında ($t=y$ anında) patlamaya ne kadar zamanın kaldığını bilmek isteriz. Bu şu anlama gelir; $t=y$ zamanında ve $y < X$ durumunda $X-y$ arasındaki zamanın istatistiksel olarak hangi dağılımdan geldiğini veya hangi dağılıma uyduğunu bilmek isteriz. Bunu da şu şekilde hesaplarız;

¹⁸ Erol Gelenbe, Guy Pujolle, **Introduction To Queueing Networks**, 2. Basım, Chichester, Wiley, 1998,s:4

$$\begin{aligned}
P\{X - y < \chi / X > y\} &= P\{y < X < y + \chi\} / P\{X > y\} \\
&= \frac{1 - e^{-\lambda(y+\chi)} - (1 - e^{-\lambda y})}{e^{-\lambda y}} \\
&= 1 - e^{-\lambda \chi} = P\{X < \chi\}
\end{aligned}$$

Bu durum bize patlamanın henüz gerçekleşmediği t zamanına kadar geçen süre olan $X-y$ süresinin dağılımının, X süresinin dağılımıyla aynı olduğunu görmemizi sağlar. Bu duruma üssel dağılımın unutkanlık özelliği (Memoryless – Markovian Property) adı verilir¹⁹.

2.7. Kuyruk Modelleri Ve Karşılaştırılmaları

Kuyruk modellerinin tanımlanmasında Kendall – Lee – Taha notasyonunun kullanıldığından yukarıda bahsetmiştik. Modeller incelemeyi amaçladıkları sistemlere göre değişiklik gösterdiklerinden bu notasyonla sınıflandırılmaları güç olacaktır. Onun yerine sistemin özelliğine göre bir tanımlama veya sınıflandırma yapmak daha doğru olur.

Bu çalışmada bu sınıflandırma modeller geliş kaynağına, servis sürelerine ve sistemdeki kanal sayılarına dayanılarak yapılacaktır.

2.7.1. Üssel Servis Zamanlı Modeller

Üssel servis zamanlı modellere en basit örnek tek servis kanallı modellerdir. Bu modellerin özelliklerini şu şekilde sıralayabiliriz²⁰:

- 1) Sisteme gelişler rassaldır ve Poisson dağılımına uyar
- 2) Servis süresi dağılımı Üssel dağılıma uyar.
- 3) Sistem tek kanallı olabileceği gibi çok ve seri kanallı da olabilir.

¹⁹ Gelenbe ve Pujolle, s: 5

²⁰ Sariaslan, s: 16

- 4) Sistemde trafik yoğunluk oranı (λ/μ) 1'den küçüktür.
- 5) Müşteriler tek tek gelir ve tek tek servis görürler. (t anında sisteme sadece 1 müşteri gelir veya sistemden sadece 1 müşteri ayrılır.)
- 6) Servis disiplini ilk gelen ilk hizmet görür şeklindedir. (Servis disiplini sistemin genel işleyişini etkilemediğinden bu modeller diğer disiplinler için de geçerlidir. Bu yüzden servis disiplini genel disiplin (GD) şeklinde gösterilecektir.)
- 7) Sistem denge durumundadır. (Eğer incelenen sistemin davranışı zamana bağlı olarak değişmiyorsa; yani zamandan bağımsızsa sistem denge durumundadır (Steady state), sistemin davranışı zamana bağlı olarak değişiyorsa sistem geçiş durumundadır (Transient state) denilir)

Burada ortalama geliş oranı λ , ortalama servis süresi μ ve trafik yoğunluk oranı (λ/μ) ρ ile gösterilecektir.

2.7.1.1. M/M/1 Modeli İçin Sistem Performans Ölçüleri

Gelişlerin Poisson, servis sürelerinin Üssel dağılıma uyduğu ve sistemde tek bir kanalın olduğu modellerde sistem performans ölçüleri ve elde edilişleri aşağıdaki gibi olacaktır.

Sistemde (Servis gören ve kuyrukta servis görmek için bekleyen) beklenen müşteri sayısı:

$$L = \frac{\rho}{1-\rho} = \frac{\lambda}{\mu-\lambda}$$

Kuyrukta beklenen müşteri sayısı:

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

Sistemde geçen (Kuyrukta ve servis süresince) ortalama zaman:

$$W = \frac{1}{(\mu - \lambda)}$$

Kuyrukta geçen ortalama zaman:

$$W_q = \frac{\lambda}{\mu(\mu - \lambda)}$$

Sistemin atıl kalma olasılığı (Sistemde hiç müşteri bulunmaması)

$$\rho_0 = 1 - \left(\frac{\lambda}{\mu} \right)$$

Sistemde n sayıda müşteri olma olasılığı

$$\rho_n = \left(\frac{\lambda}{\mu} \right)^n \rho_0 = \left(\frac{\lambda}{\mu} \right)^n \left(1 - \frac{\lambda}{\mu} \right)$$

Formüllerin nasıl elde edildiğini göstermek için herhangi bir t anını inceleyelim. Sistemin işleyişini daha iyi anlamak için t anında bu sistemde n sayıda ($n > 0$) müşteri bulunma olasılığını hesaplamaya çalışalım. Bunun için t zaman anından sonraki çok küçük bir zaman dilimi olan Δt zaman dilimindeki değişiklikleri inceleyebiliriz. Δt , sisteme en fazla bir müşterinin katılmasına veya sistemden en fazla bir müşterinin çıkmasına müsaade eden çok küçük bir zaman dilimidir. Bu zaman diliminde gerçekleşebilecek olasılıklar aşağıdaki tabloda verilmiştir²¹.

²¹ Sarıaslan, s: 18

Tablo 3**t Anında Sistemdeki Müşteri Sayısı ve Olayların Olma Olasılıkları.**

Durum	(t) Anında Müşteri Sayısı	(t) Anında Müşteri Olasılığı	Δt Zaman Aralığında Geliş	Δt Zaman Aralığında Çıkış	(t+ Δt) Zaman Anında Sistemdeki Müşteri Sayısı
1	n	$P_n(t)$	0	0	n
2	n-1	$P_{(n-1)}(t)$	1	0	(n-1)+1-0=n
3	n+1	$P_{(n+1)}(t)$	0	1	(n+1)+0-1=n
4	n	$P_{(n)}(t)$	1	1	n+1-1=n

Kaynak : Halil Sariaslan, Sıra Bekleme Sistemlerinde Simülasyon Tekniği,1986, s:18)

1. durumda t anında sistemde n sayıda müşteri varken 0 müşteri gelir ve 0 müşteri sistemden çıkar. $P_n(t)$ sistemde n sayıda müşteri bulunma olasılığını, λ sistemin geliş oranını, μ de servis oranını gösterirse Δt zaman aralığında sisteme 0 müşteri gelme olasılığı $(1 - \lambda\Delta t)$, sistemden 0 müşteri çıkma olasılığı da $(1 - \mu\Delta t)$ olacaktır. (Çünkü sisteme t anında ya 1 müşteri ya da 0 müşteri gelecektir. 1 müşterinin gelme olasılığı $\lambda\Delta t$ ise olasılık toplamı 1 olacağından 0 müşterinin gelme olasılığı $(1 - \mu\Delta t)$ olacaktır. Sistemden ayrılışlar için de aynı durum geçerlidir.). Dolayısıyla t anında sistemde n sayıda müşteri bulunma olasılığını ve sisteme 0 müşteri girişinin ve 0 müşteri çıkışının olasılığını

$$P_n(t) (1 - \lambda\Delta t) (1 - \mu\Delta t)$$

olarak gösterebiliriz.

2. durumda t anında sistemde n-1 müşteri varken Δt zaman aralığında sisteme 1 geliş ve sistemden 0 çıkış gerçekleşir. t anında sistemde n-1 müşteri bulunma olasılığı $P_{(n-1)}(t)$, sisteme 1 girişin $(\lambda\Delta t)$ ve sistemden 0 çıkışın $(1 - \mu\Delta t)$ gözlenme olasılığını

$$P_{(n-1)}(t) (\lambda\Delta t) (1 - \mu\Delta t)$$

şeklinde gösterebiliriz.

3. durumda sistemde $n+1$ sayıda müşteri varken Δt zaman aralığında sisteme 0 geliş $(1 - \lambda\Delta t)$ ve sistemde 1 çıkış $(\mu\Delta t)$ gerçekleşebilir. t anında sistemde $n+1$ sayıda müşteri bulunma olasılığı $P_{(n+1)}(t)$, sisteme 0 girişin olma olasılığı $(1 - \lambda\Delta t)$, sistemden 1 çıkışın $(\mu\Delta t)$ olma olasılığını

$$P_{(n+1)}(t) (1 - \lambda\Delta t) (\mu\Delta t)$$

olarak gösterebiliriz²².

4. durumda t anında n müşteri ve Δt zaman aralığında 1 geliş ve 1 çıkış gerçekleşebilir. t anında n sayıda müşteri bulunma olasılığı $P_{(n)}(t)$, sisteme 1 giriş $(\lambda\Delta t)$ ve sistemden 1 çıkış $(\mu\Delta t)$ gerçekleşme olasılığı

$$P_{(n)}(t) (\lambda\Delta t) (\mu\Delta t)$$

şeklinde gösterilebilir.

Bir sistemde herhangi bir $(t + \Delta t)$ anında bu dört durumdan herhangi biri kesinlikle gerçekleşecektir. Kısacası $(t + \Delta t)$ anında sistemde n sayıda müşteri bulunma olasılığı bu dört durumun olasılıkları toplamına eşittir.

$$P_{(n)}(t + \Delta t) = P_{(n)}(t) (1 - \lambda\Delta t) (1 - \mu\Delta t) + P_{(n-1)}(t) (\lambda\Delta t) (1 - \mu\Delta t) + P_{(n+1)}(t) (1 - \lambda\Delta t) (\mu\Delta t) +$$

$$P_{(n)}(t) (\lambda\Delta t) (\mu\Delta t)$$

olacaktır. Gerekli çarpma işlemleri ve sadeleştirmeler yapıldıktan sonra $(\Delta t^2, \Delta t \rightarrow 0)$ yaklaşıkça küçülür ve 0'a yaklaşır. Bu yüzden eşitlikte dikkate alınmayıp denklemden çıkarılacaktır.) aşağıdaki şekilde bulunur:

$$P_{(n)}(t + \Delta t) = P_{(n)}(t)(1 - \mu\Delta t - \lambda\Delta t) + P_{(n-1)}(t)(\lambda\Delta t) + P_{(n+1)}(t)(\mu\Delta t)^{23}$$

²² Sariaslan, s:18

²³ Sariaslan, s: 19

$P_{(n)}(t)$ değerini eşitliğin sağ tarafına geçirir ve Δt ortak terimi de parantez dışına alır ve eşitliğin her iki tarafını Δt 'ye bölersek;

$$\frac{P_n(t+\Delta t) - P_n(t)}{\Delta t} = -(\mu + \lambda)P_n(t) + \lambda P_{n-1}(t) + \mu P_{n+1}(t)$$

Δt 0'a yaklaştıkça türev kavramının tanımına göre P_n teriminin t 'ye göre türevi;

$$\frac{\partial P_n(t)}{\partial t} = \lim_{\Delta t \rightarrow 0} \frac{P_n(t+\Delta t) - P_n(t)}{\Delta t}$$

şeklinde yazılabilir. Böylece Δt 0'a yaklaşırken yukarıdaki eşitlik;

$$\frac{\partial P_n(t)}{\partial t} = -(\mu + \lambda)P_n(t) + \lambda P_{n-1}(t) + \mu P_{n+1}(t) \quad 1$$

biçiminde diferansiyel bir eşitlik olarak yazılabilir. Yukarıdaki diferansiyel denklem $n > 0$ değerleri için geliştirilmiştir. $N=0$ olursa sistemde $n-1$ müşteri bulunma olasılığı 0 olacaktır; çünkü $0-1=-1$ negatif sayıda müşterinin sistemde bulunması pek olası değildir. Böylece $P_{n-1}(t) = 0$ olarak alınıp denklemde yerine yazılırsa;

$$\frac{\partial P_0(t)}{\partial t} = -(\mu + \lambda)P_0(t) + \mu P_{0+1}(t)$$

Eşitliğin sağındaki ifadeyi açarsak,

$$\frac{\partial P_0(t)}{\partial t} = -\mu P_0(t) - \lambda P_0(t) + \mu P_1(t)$$

olur. Bu ifadede $-\mu P_0(t)$ sistemde hizmet sunulup ayrılacak olan müşteri sayısını göstermektedir. Yukarıda belirttiğimiz gibi $n=0$ olunca sistemde hiç müşteri yok demektir. Böyle olunca sistemden çıkış olası değildir. Dolayısıyla $-\mu P_0(t)$ ifadesi yukarıdaki denklemden çıkarılır.

Bu durumda $n=0$ için diferansiyel denklem:

$$\frac{\partial P_0(t)}{\partial t} = -\lambda P_0(t) + \mu P_1(t)^{24}$$

biçimine dönüştür.

Bu denklem ve daha önce elde ettiğimiz 1 nolu denklem sistemde 0, n, n-1 ve n+1 sayıda müşteri bulunma olasılığının zamanın bir fonksiyonu olduğunu ve zamana bağlı olarak değiştiğini ifade eder. Çünkü her iki denklemde de soldaki ifade türev biçiminde yazılmıştır.

$P_{(n)}(t)$ değerinin zamandan bağımsız olduğunu varsayalım. Çünkü biz olasılıkların ne olduğunu değil sistemin denge durumunu öğrenmek istiyoruz. O halde

$$\frac{\partial P_n(t)}{\partial t} = 0$$

ve

$$\frac{\partial P_0(t)}{\partial t} = 0$$

yazılabilir. Çünkü sistem denge durumunda kabul edilmekte, zamana bağlı olarak olasılık değerlerinde bir değişme olmadığı varsayılmaktadır. Bu durumda denklemlerdeki t değişkeni yok kabul edilebilir ve

$$0 = -(\mu + \lambda)P_n + \lambda P_{n-1} + \mu P_{n+1}$$

yazılabilir. Buradan

$$(\mu + \lambda)P_n = \lambda P_{n-1} + \mu P_{n+1}$$

eşitliği elde edilir. Bu eşitlikten de

$$P_{n+1} = \frac{(\lambda + \mu)P_n - \lambda P_{n-1}}{\mu}$$

²⁴ Sariaslan, s: 20

veya

$$P_{n+1} = \left(\frac{\lambda + \mu}{\mu} \right) P_n - \left(\frac{\lambda}{\mu} \right) P_{n-1} \quad 2$$

eşitliklerine ulaşılır²⁵. Aynı şekilde

$$\frac{\partial P_0}{\partial t} = 0 = -\lambda P_0 + \mu P_1$$

ifadesinden de

$$0 = -\lambda P_0 + \mu P_1$$

$$\lambda P_0 = \mu P_1$$

eşitliği elde edilir. Buradan da

$$P_1 = \left(\frac{\lambda}{\mu} \right) P_0 \quad 3$$

elde edilir. Yukarıdaki 2 nolu eşitlikte n=1 alınırsa,

$$P_2 = \left(\frac{\lambda + \mu}{\mu} \right) P_1 - \left(\frac{\lambda}{\mu} \right) P_0$$

olacaktır. 3 nolu eşitlikte P_1 değeri yerine konulursa,

$$P_2 = \left(\frac{\lambda + \mu}{\mu} \right) \left(\frac{\lambda}{\mu} \right) P_0 - \left(\frac{\lambda}{\mu} \right) P_0 = \left[\left(\frac{\lambda + \mu}{\mu} \right) \left(\frac{\lambda}{\mu} \right) - \left(\frac{\lambda}{\mu} \right) \right] P_0$$

$$P_2 = \left(\frac{\lambda^2}{\mu^2} \right) P_0 = \left(\frac{\lambda}{\mu} \right)^2 P_0 \quad 4$$

elde edilir. Aynı şekilde 2 nolu denklemde n=2 alınırsa,

$$P_3 = \left(\frac{\lambda}{\mu}\right)^3 P_0 \quad 5$$

bulunur²⁶. Benzer şekilde

$$P_n = \left(\frac{\lambda}{\mu}\right)^n P_0 \quad 6$$

ifadesi yazılabilir.

$n > 0$ ve tamsayı olacağını daha önce belirtmiştik. Bu olasılıkların toplamının 1'e eşit olacağını biliyoruz. O halde denklemi şu şekilde yazabiliriz:

$$P_0 + P_1 + P_2 + \dots + P_n + \dots = 1$$

3, 4, 5 ve 6 nolu eşitlikleri bu denklemde yerine yazarak P_0 parantezine alırsak,

$$P_0 \left[1 + \left(\frac{\lambda}{\mu}\right) + \left(\frac{\lambda}{\mu}\right)^2 + \left(\frac{\lambda}{\mu}\right)^3 + \dots + \left(\frac{\lambda}{\mu}\right)^n + \dots \right] = 1$$

$$P_0 = \left[\frac{1}{1 + \left(\frac{\lambda}{\mu}\right) + \left(\frac{\lambda}{\mu}\right)^2 + \left(\frac{\lambda}{\mu}\right)^3 + \dots + \left(\frac{\lambda}{\mu}\right)^n + \dots} \right] \quad 7$$

eşitliği bulunur²⁷. İlk bakışta bu eşitliğin paydasının sonsuza doğru artacağı; eşitliğin kendisinin de 0'a çok yakın bir sayı olacağı düşünülebilir. Halbuki $\rho = \frac{\lambda}{\mu} < 1$ olduğundan kuyruk modeli sonsuza doğru artmayacaktır. 7 nolu denklemden paydanın

²⁵ Sariaslan, s: 21

²⁶ Sariaslan, s: 22

²⁷ Sariaslan, s: 23

toplamının $a=1$, $r=\frac{\lambda}{\mu}$ olan ve $S = \frac{a}{1-r}$ şeklinde hesaplanan bir geometrik dizi olduğunu görebiliriz. Şimdi bu değeri yerine yazarsak,

$$P_0 = \frac{1}{1 - \left(\frac{\lambda}{\mu}\right)}$$

buradan da,

$$P_0 = 1 - \left(\frac{\lambda}{\mu}\right)$$

bulunur. Bu değeri 6 nolu denklemde yerine yazarsak,

$$P_n = \left(\frac{\lambda}{\mu}\right)^n \left(1 - \frac{\lambda}{\mu}\right) \quad 8$$

olacaktır. Bu basit denklem sistemde n ($n>0$) sayıda müşteri bulunmasının olasılığını verir. Buradan hareketle sistemde bulunacak toplam müşteri sayısını (L) şu şekilde hesaplayabiliriz;

$$L = P_0 n_0 + P_1 n_1 + P_2 n_2 + \dots + P_n n_n + \dots = \sum_{n=0}^{\infty} P_n(n) \text{ , dir}^{28}.$$

8 nolu denklemdeki P_n değerini yerine koyarsak,

$$L = \sum_{n=0}^{\infty} \left(\frac{\lambda}{\mu}\right)^n \left(1 - \frac{\lambda}{\mu}\right) (n) = \left(1 - \frac{\lambda}{\mu}\right) \sum_{n=0}^{\infty} \left(\frac{\lambda}{\mu}\right)^n (n) \quad 9$$

elde edilir. Buradan kolaylıkla görülebileceği gibi

²⁸ Sariaslan, s: 25

$\sum_{n=0}^{\infty} (n) \left(\frac{\lambda}{\mu} \right)^n$ ifadesi $a = \frac{\lambda}{\mu}$ olacak şekilde $0, a, 2a^2, 3a^3, \dots, na^n, \dots$ sonsuza

doğru giden bir dizidir. Bu dizinin toplamını,

$S = \frac{a}{(1-a)^2}$ bağlantısıyla elde edebiliriz. a değerini yerine koyarsak;

$$\sum_{n=0}^{\infty} (n) \left(\frac{\lambda}{\mu} \right)^n = \frac{\frac{\lambda}{\mu}}{\left(1 - \frac{\lambda}{\mu} \right)^2}$$

olacaktır. 9 nolu eşitlikte bu değerleri yerine koyarsak,

$$L = \left(1 - \frac{\lambda}{\mu} \right) \left(\frac{\frac{\lambda}{\mu}}{\left(1 - \frac{\lambda}{\mu} \right)^2} \right)$$

$$L = \frac{\lambda}{\mu - \lambda}$$

10

olarak bulunur²⁹. Bu bağıntıdan hareketle sisteme ait diğer göstergelerin çıkarımları yapılabilir. Örneğin sistemdeki müşteri sayısı ve kuyrukta bekleyen müşteri sayısı ayrı ayrı hesaplanabilir. Sistemdeki olması beklenen toplam müşteri sayısından kuyruktaki beklenen müşteri sayısını çıkarırsak hizmet görmekte olan müşterilerin beklenen sayısını elde edebiliriz.

$$L - L_q = 1 - \left(1 - \frac{\lambda}{\mu} \right) = \frac{\lambda}{\mu}$$

Buradan hareketle kuyrukta olması beklenen müşteri sayısını hesaplayabiliriz.

²⁹ Sariaslan, s: 25

$$L_q = \left(\frac{\lambda}{\mu - \lambda} \right) - \left(\frac{\lambda}{\mu} \right) = \left(\frac{\lambda^2}{\mu(\mu - \lambda)} \right)$$

bulunur.

2.7.1.1.1. Little Formülü

Rassal bir kuyruk sürecinin oluştuğunu düşünelim. İlk başta bu kuyruğun yapısı önemli değildir; bizi ilgilendiren sistemin denge durumunda olması, kuyruğa belli sayıda müşterilerin katılması, belli bir süre kuyrukta kalması ve sonra belli bir sürede hizmet görüp kuyruktan ayrılmalarıdır. L sistemde beklenen müşteri sayısı W sistemde beklenen ortalama süresi olmak üzere Little aşağıdaki formülü önermiştir³⁰;

$$L = \lambda W$$

Literatürde çok sık kullanılan bu formül, sistemdeki ortalama müşteri sayısının müşterilerin ortalama geliş hızlarıyla sistemde harcadıkları ortalama zamanın çarpımına eşit olduğunu varsayar.

Buradan hareketle 10 nolu eşitlikten,

$$L = \lambda W$$

$$W = \frac{L}{\lambda}$$

$$W = \frac{1}{\mu - \lambda}$$

eşitlikleri elde edilir. Buradan hareketle kuyrukta ortalama bekleme zamanını da hesaplayabiliriz. Bir müşterinin ortalama bekleme zamanı W'den hizmet süresince geçmesi beklenen ortalama zamanı çıkarırsak (μ servis oranı birim zamanda hizmet sunulan müşteri sayısıysa bir müşteri için harcanan ortalama süre $\frac{1}{\mu}$ olacaktır.)

³⁰ Mitrani, s: 95

$$W_q = W - \frac{1}{\mu}$$

$$W_q = \frac{\lambda}{\mu(\mu - \lambda)}$$

olarak elde edilir³¹.

2.7.1.2. M/M/c Modeli İçin Sistem Performans Ölçüleri

Bu modelin yukarıdaki modelden tek farkı bir yerine birden fazla paralel haldeki hizmet sunucularının olmasıdır. Daha önce de hatırlatması yapıldığı gibi her bir hizmet sunucusunun ortalama servis süresinin aynı olduğu varsayımı altında model incelenecektir.

c kanallı bir kuyruk modeli için trafik yoğunluk oranı :

$$\rho = \frac{\lambda}{c\mu} < 1 \text{ şeklinde gösterilir.}$$

Diğer performans ölçütleri ise şu şekildedir:

Sistemin boş kalma olasılığı:

$$P_0 = \left[\left(\sum_{n=0}^{c-1} \frac{1}{n!} \left(\frac{\lambda}{\mu} \right)^n \right) + \left(\frac{1}{c!} \left(\frac{\lambda}{\mu} \right)^c \left(\frac{c\mu}{c\mu - \lambda} \right) \right) \right]^{-1}$$

Sistemde n sayıda müşteri olma olasılığı³²:

$$P_n = \begin{cases} P_0 \frac{(c\rho)^n}{n!} & ; n < c \\ P_0 \frac{\rho^n c^c}{c!} & ; n \geq c \end{cases} \quad \text{veya} \quad P_n = \begin{cases} \frac{\lambda^n}{n! \mu^n} P_0 & , n < c \\ \frac{\lambda^n}{c^{n-c} c! \mu^n} P_0 & , n \geq c \end{cases}$$

³¹ Mitrani, s: 95

³² Andreas Willig, *A Short Introduction To Queueing Theory*, Technical University Berlin, 21 Temmuz 1999, Berlin – Almanya, s: 14

Kuyrukta beklenen müşteri sayısı:

$$L_q = \frac{\lambda \mu \left(\frac{\lambda}{\mu} \right)^c}{(c-1)!(c\mu - \lambda)^2} P_0$$

Sistemdeki beklenen müşteri sayısı:

$$L = L_q + \frac{\lambda}{\mu}$$

Kuyrukta harcanan ortalama zaman:

$$W_q = \frac{L_q}{\lambda}$$

Sistemde harcanan ortalama zaman:

$$W = W_q + \frac{1}{\mu}$$

şeklinde elde edilir.

2.7.1.3. M/M/c/K Modeli İçin Sistem Performans Ölçüleri

M/M/c/K kuyruk modeli, müşterilerin sisteme gelişlerinin Poisson dağılımına, servis sürelerinin Üssel dağılıma uyduğu çok kanallı bir modeldir. Daha önce incelenen modellerden farkı bu modelin K sayıdaki müşteriye hizmet edebilecek kapasitede olmasıdır. Yani sistem sadece K sayıdaki müşteriye hizmet verebilir. Eğer sistemde K sayıda müşteri varsa yeni gelen müşteriler kuyruğa dahil edilmeden sistemden ayrılırlar veya sistemden daha önce gelen müşterilerin servis görüp ayrılmasını beklerler. Bu modelde sistem performans ölçülerine yaklaşım M/M/c modeliyle benzerlik gösterir. Tek farkı $c \geq K$ olması durumunda λ 0'a eşit olacaktır.

Denge durumu olasılıkları ve performans ölçütleri aşağıda verilmiştir. ($r = \frac{\lambda}{\mu}$)

Sistemin boş kalma olasılığı:

$$P_0 = \begin{cases} \left(\sum_{n=0}^{c-1} \frac{r^n}{n!} + \frac{r^c}{c!} \frac{1-\rho^{K-c+1}}{1-\rho} \right)^{-1} & (\rho \neq 1), \\ \left(\sum_{n=0}^{c-1} \frac{r^n}{n!} + \frac{r^c}{c!} (K-c+1) \right)^{-1} & (\rho=1) \end{cases} \quad 11$$

Sistemde n sayıda müşteri olma olasılığı:

$$P_n = \begin{cases} \frac{\lambda^n}{n! \mu^n} P_0 & (1 \leq n < c) \\ \frac{\lambda^n}{c^{n-c} c! \mu^n} P_0 & (c \leq n \leq K) \end{cases} \quad 12$$

Kuyrukta beklenen müşteri sayısı:

$$L_q = \frac{P_0 r^c \rho}{c! (1-\rho)^2} \left[1 - \rho^{K-c+1} - (1-\rho)(K-c+1)\rho^{K-c} \right]$$

Sistemde beklenen müşteri sayısı :

$$L = L_q + \frac{\lambda(1-P_K)}{\mu} = L_q + r(1-P_K)$$

Sistemde harcanan ortalama zaman:

$$W = \frac{L}{\lambda(1-P_K)}$$

Kuyrukta harcanan ortalama zaman:

$$W_q = W - \frac{1}{\mu}$$

Bu modele özgü olarak $\rho > 1$ durumda da sistemin denge durumundadır³³. Çünkü K kısıtı konmasaydı kuyruk tanımlanamayacak şekilde hızla büyüyecek ve modelleme yapılamayacaktı. Fakat bu modelde sisteme en fazla K sayıda müşteri gelişine izin verildiği için kuyruğun bu belirsiz artışının önüne geçilmiştir.

Kısıtın konulması formülasyonu kolaylaştırıyor gibi görünse de sistemin performans ölçülerinin hesaplanmasında bazı düzeltmeler yapılmasına gidilmiştir. Örneğin λ yerine λ_{eff} kullanılır. Bu efektif varış anlamına gelir ve aslında λ parametresi üzerinde düzeltme yapıldığını ifade eder. Çünkü sistemde N müşteri varsa K kısıtı konulduğunda müşteri kaybına neden olunacaktır. Bunu ortadan kaldırmak için $\lambda_{eff} = \lambda(1 - P_K)$ düzeltme işlemi yapılarak efektif varış oranı hesaplanabilir³⁴.

2.7.1.4. M/M/1/K Modeli İçin Sistem Performans Ölçüleri

Varış sürecinin Poisson, servis süresinin üssel olduğu tek kanallı kuyruk modellerinde K sayıdaki müşteriye hizmet verecek kapasitedeki modeller için tanımlanmıştır. Yukarıdaki modelden tek farkı $c=1$ olduğu durumlar için geçerlidir. Sistem performans ölçüleri aşağıdaki şekilde hesaplanır;

Sistemde n sayıda müşteri olma olasılığı:

$$P_n = \begin{cases} \frac{\rho^n (1 - \rho)}{1 - \rho^{K+1}} & , \rho \neq 1 \\ \frac{1}{K+1} & , \rho = 1 \end{cases}$$

Sistemin boş olma olasılığı³⁵:

$$P_0 = \begin{cases} \frac{1 - \rho}{1 - \rho^{K+1}} & , \rho \neq 1 \\ \frac{1}{K+1} & , \rho = 1 \end{cases}$$

³³ Willig, s: 16

³⁴ Gross ve Haris, s: 76

³⁵ Gross ve Haris, s: 77

Sistem performans ölçüleri hesaplanırken öncelikle yukarıdaki model için tahmin edilen λ_{eff} katsayısı hesaplanır.

$$\lambda_{eff} = \begin{cases} \frac{\lambda(1-\rho^K)}{1-\rho^{K+1}} & , \rho \neq 1 \\ \frac{\lambda K}{K+1} & , \rho = 1 \end{cases}$$

Buna göre sistemde beklenen müşteri sayısı :

$$L = \frac{\rho}{1-\rho} - \frac{(K+1)\rho^{K+1}}{1-\rho^{K+1}}$$

Kuyrukta beklenen müşteri sayısı:

$$L_q = L - \frac{\lambda_{eff}}{\mu} L - \frac{\rho(1-\rho^K)}{1-\rho^{K+1}}$$

Sistemde harcanan ortalama zaman:

$$W = \frac{L}{\lambda(1-P_K)}$$

Kuyrukta harcanan ortalama zaman:

$$W_q = \frac{L_q}{\lambda(1-P_K)}$$

olacak şekilde elde edilir.

2.7.1.5. M/M/c/c Modeli İçin Sistem Performans Ölçüleri (Erlang Formülü)

M/M/c/K modellerinde K=c olması durumunda karşılaşılan kuyruk modelidir. Sisteme katılmasına müsaade edilen müşteri sayısı sistemdeki kanal sayısına eşitse yeni kuyruk oluşumuna izin verilmez veya mevcut kuyruklardan birine herhangi bir müşteri katılmasına izin verilmez.

$K=c$ olması durumunda 11 ve 12 nolu denklemlerinden hareketle aşağıdaki eşitlikler yazılabilir³⁶;

$$P_n = \frac{\frac{(\lambda / \mu)^n}{n!}}{\sum_{n=0}^c \frac{(\lambda / \mu)^n}{n!}}, \quad (0 \leq n \leq c) \quad 13$$

Burada sistemin c sayıdaki müşteriye hizmet etmesi kısıdıyla parametrelerin tahmininde doğabilecek yanlışlığı düzeltmek için Earlang Formülü kullanılır. Bu formülün kullanılmasıyla denge durumundaki sistemin tamamen dolu olması halinde elde edilecek parametreler elde edilir³⁷.

$$P_c = \frac{\frac{r^c}{c!}}{\sum_{n=0}^c \frac{r^n}{n!}}, \quad (r=\lambda/\mu)$$

$$\lambda_{eff} = \lambda(1 - P_c)$$

Sistemde ortalama harcanan zaman

$$W = \frac{1}{\mu}$$

Sistemde beklenen müşteri sayısı

$$L = \lambda_{eff} W = \frac{\lambda(1 - P_c)}{\mu}$$

c sayıda kanal ve c sayıda müşteri kısıtı olduğundan kuyrukta ve sistemde bekleme durumuyla karşılaşılmaz. O halde aşağıdaki eşitliği yazabiliriz;

$$L_q = W_q = 0$$

³⁶ Gross ve Haris, s: 80

³⁷ Gross ve Haris, s: 80

Bu formül aslında Kuyruk Teorisinin ilk kullanılmaya başlandığı halidir. M/M/c/c modeli Earlang'ın 1917 yılında telefon konuşmalarını dikkate alarak kurduğu modeldir ve basit bir telekomünikasyon ağını analiz eder³⁸.

Yapılan aramaların sisteme giriş süreci Poisson, servis süreleri üssel, tesadüfi değerlerden oluşan sistemi tanımlar. Sistem doluysa, diğer bir deyişle c sayıdaki kanalın hepsi meşgulse meşgul sinyali alınır. Bu model servis sürelerinin deterministik ve bağımsız olduğu modeller için de aynıdır.

2.7.1.6. M/M/∞ Modeli İçin Sistem Performans Ölçüleri

Sınırsız sayıda sunucudan oluşan bir sistem düşünülmesi zor olabilir ama hayatta sıkça karşılaşılan bir durumdur. Self-servis de denilen fast food restoranları bu modele örnek olarak gösterilebilir.

Model için aşağıdaki eşitlikler elde edilmiştir³⁹.

$$P_0 = \left(\sum_{n=0}^{\infty} \frac{r^n}{n!} \right)^{-1}$$

$$P_n = \frac{r^n}{n!} P_0 = \frac{r^n e^{-r}}{n!}, \quad (n \geq 0) \quad 14$$

Yukarıdaki eşitlikten anlaşılacağı gibi denge durumundaki sistemin olasılık fonksiyonu aslında $r = \lambda / \mu$ parametrelili Poisson dağılımıdır. Sistemde beklenen müşteri sayısı 14 nolu denklemin ortalamasına, $r = \lambda / \mu$ parametrelili Poisson dağılımının ortalamasına eşittir.

$$L = r = \frac{\lambda}{\mu}$$

Sistemde sonsuz sayıda kanal olduğu için kuyrukta bekleyen müşteri sayısı 0 olacağından;

³⁸ Gross ve Harris, s: 81

³⁹ Gross ve Harris, s: 81

$$L_q = W_q = 0$$

yazılabilir. Sistemde ortalama bekleme süresinin sistemdeki ortalama servis süresine eşit olduğu ise açıktır.

$$W = \frac{1}{\mu}$$

2.7.1.7. PASTA Teoremi

Poisson varış süreçlerine sahip modellerin kendine özgü bir özelliği vardır. Sistemin herhangi bir t anında müşteri sayısını incelersek, incelediğimiz t anındaki müşterilerin sahip olduğu istatistiksel özelliklerle modelin geneli için geçerli olan istatistiklerin aynı olduğu kabul edilir. Bu duruma PASTA Teoremi (Poisson Arrivals See Time Averages) adı verilir. Bu durum deterministik varış süresine veya deterministik servis süresine sahip modeller için geçerli değildir. Çünkü bu modeller otomasyona dayalı oldukları için genellikle kuyrukta bekleyen müşteri olmayacaktır⁴⁰.

2.7.2. Earlang Servis Zamanlı Kuyruk Modeli

2.7.2.1. M/E_k/1 Modeli İçin Sistem Performans Ölçüleri

Çoğu uygulamada Poisson varış sürecinin geçerli olmasına rağmen üssel servis sürecinin geçerli olmadığı görülebilir. Bu tarz durumlarda Earlang modelin kurulması için gamma dağılımını önermiştir. Gamma dağılımı k simgesi ile belirtilen bir parametreye göre şekil alabilen bir dağılımdır. Aynı zamanda üssel veya sabit servis zamanlı dağılımlar gamma dağılımının özel bir biçimidir⁴¹.

Bu dağılım sayesinde tek kanallı seri haldeki kuyrukların modellenmesi de yapılabilir. Burada önemli olan varış sürecinin Poisson, servis süresinin üssel olduğu k sayıdaki seri halde kuyruğun tanımlanmış olmasıdır⁴².

⁴⁰ Willig, s: 21

⁴¹ Sariaslan, s: 32

⁴² Ivo Adan, Jacques Resing, **Queueing Theory**, Eindhoven – Hollanda, Eindhoven University Of Technology, 14 Şubat 2001, s: 49

$k=1$ olduğu durumda servis süresi üssel dağılımı, $k=\infty$ olduğu durumda ise sabit dağılımı tanımlar. Sistem performans değerlerinin ölçülmesi için M/M/1 modelinin değerleri üzerinde düzeltme yapılması yeterli olacaktır.

$$\rho = \lambda \frac{k}{\mu}$$

$$W = \left(\frac{\rho}{1-\rho} \right) \left(\frac{k+1}{2} \right) \left(\frac{1}{\mu} \right)$$

Little formülü yardımıyla aşağıdaki değişkenler elde edilir;

$$W_q = W - \frac{1}{\mu} = \left(\frac{\rho}{1-\rho} \right) \left(\frac{k+1}{2} \right) \left(\frac{1}{\mu} \right) - \left(\frac{1}{\mu} \right)$$

$$L = \lambda W = \lambda \left(\frac{\rho}{1-\rho} \right) \left(\frac{k+1}{2} \right) \left(\frac{1}{\mu} \right)$$

$$L_q = L - \frac{\lambda}{\mu} = \lambda \left(\frac{\rho}{1-\rho} \right) \left(\frac{k+1}{2} \right) \left(\frac{1}{\mu} \right) - \left(\frac{\lambda}{\mu} \right)$$

Burada $k=1$ olursa model M/M/1 modeliyle aynı özelliklere sahip olacaktır. $k=\infty$ olması durumundaysa Poisson varış sürecine sahip deterministik servis süreciyle tanımlanan model elde edilecektir. Bu durumda da gamma, üssel ve deterministik model arasındaki ilişki daha iyi bir şekilde anlaşılabilir⁴³.

Modelde kanal sayısının artması formülasyonu güçleştireceğinden bu tarz uygulamalar daha çok bilgisayar programları yardımıyla çözülebilmektedir.

⁴³ Sarıaslan, s: 33

2.7.3. Deterministik Servis Zamanlı Kuyruk Modeli

2.7.3.1. M/D/1 Modeli İçin Sistem Performans Ölçüleri

Çoğu uygulamada sisteme varış sürecinin Poisson, ortalama servis süresinin deterministik, yani sabit olduğu tek kanallı kuyruk modelleriyle karşılaşabiliriz. Bu model aslında fabrikasyon üretime dayanan ve makinelerle üretimin yapıldığı atölyeler için kullanılan bir modeldir. Ortalama servis süresinin sabit olduğu ve her bir müşteriye aynı sürede servis yapıldığı açıktır. Bu modellerde üssel servis zamanlı modellere göre sistem performans ölçülerinin yarı yarıya düştüğü kanıtlanmıştır⁴⁴. (Sistem performans ölçülerinin M/M/1 modeline ait değerlerin yarısı olacağından ayrıca gösterilmemiştir.)

2.7.4. Genel Servis Zamanlı Kuyruk Modeli

2.7.4.1. M/G/1 Modeli İçin Sistem Performans Ölçüleri

Buraya kadar incelenen modeller geliş sürecinin Poisson, servis süresinin ise üssel olduğunu varsaymaktaydı. Earlang dağılımında olduğu gibi burada da servis süresine ait bu kısıtı ortadan kaldıracak yeni bir yöntem geliştirilmiştir. Bu yöntem yine geliş sürecinin Poisson olduğu varsayımı altında servis süresinin hiçbir dağılıma bağlı kalmadan kuyruk modelinin analizini sağlar. Bunun için Pollaczek – Khintchine formülü kullanılmaktadır. Bu formülde farklı olarak servis sürelerinin dağılımının varyansı (δ^2) kullanılmıştır. Buna göre sistem performans ölçüleri de aşağıdaki şekilde hesaplanır⁴⁵;

Kuyrukta beklenen müşteri sayısı;

$$L_q = \frac{\left(\frac{\lambda}{\mu}\right)^2 + \lambda^2 \delta^2}{2\left(1 - \frac{\lambda}{\mu}\right)}$$

⁴⁴ Sariaslan, s: 31

⁴⁵ Sariaslan, s: 34

Sistemde beklenen müşteri sayısı;

$$L = \frac{\lambda}{\mu} + L_q = \frac{\lambda}{\mu} + \frac{\left(\frac{\lambda}{\mu}\right)^2 + \lambda^2 \delta^2}{2\left(1 - \frac{\lambda}{\mu}\right)}$$

Sistemde ortalama bekleme süresi;

$$W = \frac{L}{\lambda} = \frac{1}{\mu} + \frac{\left(\frac{\lambda}{\mu^2}\right) + \lambda \delta^2}{2\left(1 - \frac{\lambda}{\mu}\right)}$$

Kuyrukta ortalama bekleme süresi;

$$W_q = \frac{L_q}{\lambda} = \frac{\left(\frac{\lambda}{\mu^2}\right) + \lambda \delta^2}{2\left(1 - \frac{\lambda}{\mu}\right)}$$

2.7.5. Yığınsal Kuyruk Modelleri

Bundan önceki tüm kuyruk modellerinin dayandığı temel varsayım müşterilerin tek tek geldiği ve tek tek servis gördüğüydü. Uygulamada bu varsayımın geçerli olmadığı çok sıra bekleme sistemi vardır. Ulaşım sistemlerinde bir belediye otobüsünün durakta bekleyen bir grup yolcuyu aynı anda alması, hava alanına inen bir uçağın hava alanındaki pasaport, bagaj, vb. servis birimlerine bir grup müşteriyi aynı anda getirmesi böyle sistemlere örnek verilebilir.

Gelişlerin ya da servisin yığınsal özellik gösterme durumuna göre de iki tür model geliştirilmiştir. Bu modellerden Poisson Yığınsal Gelişler Modeli (Poisson Bulk Arrivals) gelişlerin rassal olarak büyüklüğü değişen gruplar halinde Poisson dağılımlı aralıklarla olacağı ve müşterilerin üssel servis zaman dağılımlı tek kanaldan tek tek

servis edileceği varsayımına dayanmaktadır. Bu durumda sistemin trafik yoğunluk oranı;

$$\rho = \frac{\lambda E(b)}{\mu}$$

olmakta ve grup büyüklüğünün ortalamasını temsil eden $E(b)$ 'nin değeri 1 olduğu zaman model M/M/1 modeli biçimini almaktadır⁴⁶.

Diğer yandan Üssel Yığın Servis Modeli (Exponential Bulk Service) de gelişlerin tek tek ve servisin yığınsal olarak üssel dağılımlı aralıklarla tek kanaldan olacağını varsaymaktadır.

Böylece bu iki yığınsal kuyruk modelleri uygulamada karşılaşılabilen yığınsal geliş ve yığınsal servis gibi iki özel duruma belli varsayımlara dayalı olarak çözüm getirmektedir. Ancak gelişlerin Poisson, servis zamanının üssel dağılım göstermediği ya da çok kanalın bulunmaması gibi durumlarda sorunun matematiksel formüllerle çözümü çok karmaşıklaşacağından böyle karmaşık sistemlerde çözüm için simülasyon yöntemi önerilmektedir⁴⁷.

2.8. Doğum – Ölüm Süreci

Kuyruk modellerinin pek çoğu sisteme varışların ve sistemden ayrılışların doğum – ölüm sürecine (Birth – Death Process) göre gerçekleştiğini varsayar. Bu süreç olasılık teorisinde çok fazla alanda uygulama şansı bulmaktadır.

Kuyruk modelleri için düşünecek olursak, doğum sisteme yeni bir müşterinin katılması, ölüm ise servis gören müşterinin sistemden ayrılması anlamına gelmektedir. Sistemin t anındaki durumunu ($t \geq 0$), $N(t)$ ile gösterelim. $N(t)$, t anında sistemde N sayıda müşteri olduğu anlamına gelecektir. Doğum – ölüm süreci t'nin değişmesi halinde $N(t)$ 'nin olasılıksal olarak nasıl değişeceğini gösterecektir. Daha genel bir tanım yapmak gerekirse doğum ve ölüm olaylarının (Sisteme varış ve sistemden ayrılışların) ortalama gerçekleşme zamanlarının içinde bulunduğu duruma bağlı (t)

⁴⁶ Sariaslan, s: 35

⁴⁷ Sariaslan, s: 36

olarak gerçekleşmelerine rağmen; doğum ve ölüm olayları tamamıyla rassal olarak gerçekleşmektedir. Bunun için aşağıdaki varsayımlara göz atalım:

Varsayım 1: Verilen bir $N(t) = n$ durumunda yeni bir doğum olayına kadar geçen sürenin dağılımı λ_n parametresiyle gösterilen Üssel dağılıma uyar ($n=0,1,2,\dots$)

Varsayım 2: Verilen bir $N(t) = n$ durumunda yeni bir ölüm olayına kadar geçen sürenin dağılımı μ_n parametresiyle gösterilen Üssel dağılıma uyar ($n=0,1,2,\dots$)

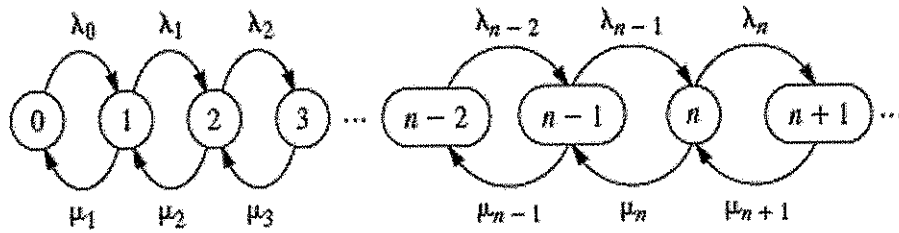
Varsayım 3: 1. ve 2. varsayımlarda gösterilen doğum ve ölüm olaylarının her ikisi de bağımsız gerçekleşir. Sistemdeki bir sonraki geçiş durumları;

$n \rightarrow n+1$ (Tek bir doğumun gerçekleşmesi)

$n \rightarrow n-1$ (Tek bir ölümün gerçekleşmesi)

olarak gösterilebilir⁴⁸.

Bu varsayımlara bakarak bu sürecin sürekli Markov Zincirlerinin özel bir hali olduğunu söyleyebiliriz.



Şekil 4. Doğum – Ölüm Süreci

Yukarıdaki şekilde oklar sistemde, bir durumdan diğer duruma geçişlerin mümkün olduğu durumları göstermektedir.

⁴⁸ Frederick S. Hillier ve Gerald J. Lieberman, **Introduction To Operations Research**, 7. Basım, Boston – ABD, The McGraw – Hill Companies Inc. , 2001, s: 848

Bazı özel durumlar hariç doğum ölüm sürecinin geçiş durumunda analizi zor olabilir. $N(t)$ olasılık dağılımına ait bazı sonuçlar elde edilebilir ama uygulamalarının yapılması karmaşık ve zordur. Diğer taraftan sistem denge durumuna geldikten sonra $N(t)$ olasılık dağılımının sonuçlarına ait çıkarımlar ve uygulamaları rahatlıkla yapılabilir.

Aşağıdaki durumu inceleyelim;

$E_n(t)$ = t anında sistemin n durumuna geçişlerinin sayısı

$L_n(t)$ = t anında sistemin n durumundan çıkışlarının sayısı

Bu iki tip olay birbirinin alternatifidir; sisteme bir kişi girer veya sistemden bir kişi çıkar. Bu olayların sayısı birbirine eşittir veya aralarında sadece 1 fark bulunur. Buradan aşağıdaki çıkarımları elde edebiliriz⁴⁹;

$$|E_n(t) - L_n(t)| \leq 1$$

$$\left| \frac{E_n(t)}{t} - \frac{L_n(t)}{t} \right| \leq \frac{1}{t}$$

$$\lim_{t \rightarrow \infty} \left| \frac{E_n(t)}{t} - \frac{L_n(t)}{t} \right| = 0$$

$E_n(t)$ ve $L_n(t)$ sayılarını t'ye bölerek bu iki olayın her bir zamanda gerçekleşme sayısını, limitini alarak da ortalama gerçekleşme zamanlarını buluruz.

$$\lim_{t \rightarrow \infty} \frac{E_n(t)}{t} = \text{Sürecin n durumuna girdiği olayların ortalama sayısı}$$

$$\lim_{t \rightarrow \infty} \frac{L_n(t)}{t} = \text{Sürecin n durumundan çıktığı olayların ortalama sayısı}$$

⁴⁹ Hillier ve Lieberman, s: 849

Buradan řu sonuca ulařırız; sistemin herhangi bir n durumunda ($n=0,1,2,\dots$) sisteme ortalama varıř oranı sistemi ortalama terk ediř oranına eřittir (Rate in = Rate out Principle)⁵⁰.

Bu eřitlik n durumundaki sũreç iin denge denklemi olarak adlandırılır. Denge denklemlerini oluřturduktan sonra her bir durum iin bu denklemleri yerine koyarak olasılık deęerlerini hesaplayabiliriz.

Sistemin 0 durumunda olduęunu dũřnelim. Sũreç 0 durumuna sadece ve sadece 1 durumundan geebilir. Dięer bir deyiřle sistem denge durumundayken sũrecin 1 durumunda olma olasılıęı (P_1), sũrecin 0 durumuna geebilmesi muhtemel durumları gũsterir. Sistemin 1 durumunda olması halinde, sistemin 0 durumuna ortalama geiř oranı μ_1 'dir. dięer tũm durumlar iin bu oran 0'a eřittir. Bu yũzden sũre boyunca sistemin bulunduęu durumdan 0 durumuna geiřinin olasılıęını řu řekilde gũsterebiliriz:

$$\mu_1 P_1 + 0(1 - P_1) = \mu_1 P_1$$

Aynı dũřũnceyle ortalama terk ediř oranını da $\lambda_0 P_0$ olarak gũsterebiliriz. O halde sistemin 0 durumundayken denge denklemini řu řekilde yazabiliriz:

$$\mu_1 P_1 = \lambda_0 P_0$$

Sũrecin bũtũn durumları iin denge denklemlerini ise řu řekilde oluřturabiliriz:

⁵⁰ Hillier ve Lieberman, s: 850

Tablo 4
Doğum – Ölüm Sürecinde Denge Durumu Denklemleri

Durum	Denge Denklemleri
0	$\mu_1 P_1 = \lambda_0 P_0$
1	$\lambda_0 P_0 + \mu_2 P_2 = (\lambda_1 + \mu_1) P_1$
2	$\lambda_1 P_1 + \mu_3 P_3 = (\lambda_2 + \mu_2) P_2$
....	
n-1	$\lambda_{n-2} P_{n-2} + \mu_n P_n = (\lambda_{n-1} + \mu_{n-1}) P_{n-1}$
n	$\lambda_{n-1} P_{n-1} + \mu_{n+1} P_{n+1} = (\lambda_n + \mu_n) P_n$
...	...

Kaynak : F. S. Hillier, G. J. Lieberman, McGraw-Hill, Introduction to Operations Research, 2001, s:850

Bu denklemler yardımıyla olasılık değerlerini aşağıdaki şekilde elde edebiliriz⁵¹:

Tablo 5
Denge Durumu Denklemlerinden Olasılık Değerlerinin Elde Edilişi

Durum	Olasılık Değerleri		
0	$P_1 = \frac{\lambda_0}{\mu_1} P_0$		
1	$P_2 = \frac{\lambda_1}{\mu_2} P_1 + \frac{1}{\mu_2} (\mu_1 P_1 - \lambda_0 P_0)$	$= \frac{\lambda_1}{\mu_2} P_1$	$= \frac{\lambda_1 \lambda_0}{\mu_2 \mu_1} P_0$
2	$P_3 = \frac{\lambda_2}{\mu_3} P_2 + \frac{1}{\mu_3} (\mu_2 P_2 - \lambda_1 P_1)$	$= \frac{\lambda_2}{\mu_3} P_2$	$= \frac{\lambda_2 \lambda_1 \lambda_0}{\mu_3 \mu_2 \mu_1} P_0$
...	...	...	...
n-1	$P_n = \frac{\lambda_{n-1}}{\mu_n} P_{n-1} + \frac{1}{\mu_n} (\mu_{n-1} P_{n-1} - \lambda_{n-2} P_{n-2})$	$= \frac{\lambda_{n-1}}{\mu_n} P_{n-1}$	$= \frac{\lambda_{n-1} \lambda_{n-2} \dots \lambda_0}{\mu_n \mu_{n-1} \dots \mu_1} P_0$
n	$P_{n+1} = \frac{\lambda_n}{\mu_{n+1}} P_n + \frac{1}{\mu_{n+1}} (\mu_n P_n - \lambda_{n-1} P_{n-1})$	$= \frac{\lambda_n}{\mu_{n+1}} P_n$	$= \frac{\lambda_n \lambda_{n-1} \dots \lambda_0}{\mu_{n+1} \mu_n \dots \mu_1} P_0$
...	...	...	...

Kaynak : F. S. Hillier, G. J. Lieberman, s: 851

⁵¹ Hillier ve Lieberman, s: 851

2.9. Öncelikli Servis Disiplini

Öncelikli servis disiplini iki biçimde uygulanır. Birincisinde, öncelikli bir müşteri gelir ve kanalda hizmet sunulan bir müşteri varsa kanaldaki müşteri çıkarılır, yerine gelen öncelikli müşteri alınır. Buna Tam Öncelikli Servis Disiplini (Preemptive Service Priority Order – PSPO) adı verilir. Eğer kanaldaki müşterinin hizmeti bitirilir ve öncelikli müşteri sırada bekleyenlerden önce alınırsa buna da Tam Öncelikli Olmayan Servis Disiplini (Non – Preemptive Priority Service – NPPS) denir⁵².

Her iki modelde de farklı öncelik sıralamasına sahip müşterilerin (Veya sınıfların) varlığı göz ardı edilmiştir. Üssel dağılım özelliğinden dolayı müşterilerin sisteme gelişlerinin dağılımı Poisson dağılımına, servis sürelerinin dağılımı da üssel dağılıma uymaktadır. Bu yönüyle öncelik sırasının bulunduğu modeller M/M/c modelleriyle benzerlik göstermektedir. Dolayısıyla bu modeller için kullanılan varsayımlar öncelikli servis disiplininin geçerli olduğu modellere de uygulanabilmektedir⁵³.

2.9.1. Çok Kanallı Tam Öncelikli Olmayan Modeller İçin Sistem Performans Ölçüleri

W_k : k sınıfına ait tam öncelikli model için ortalama sistemde bekleme süresi

$$W_k = \frac{1}{AB_{k-1}B_k} + \frac{1}{\mu}, \quad k = 1, 2, \dots, N$$

$$L_k = \lambda_k W_k, \quad k = 1, 2, \dots, N$$

$$A = c! \frac{c\mu - \lambda}{r^s} \sum_{j=0}^{s-1} \frac{r^j}{j!} + c\mu$$

⁵² Sarıaslan, s: 11

⁵³ Hillier ve Lieberman, s: 880

$$B_0 = 1$$

$$B_k = 1 - \frac{\sum_{i=1}^k \lambda_i}{c\mu}$$

c = Sistemdeki kanal sayısı

μ = Ortalama servis süresi

λ_i = i Öncelik sınıfına ait müşterilerin ortalama geliş oranı

$$\lambda = \sum_{i=1}^N \lambda_i$$

$$r = \frac{\lambda}{\mu}$$

Model, $\sum_{i=1}^k \lambda_i < c\mu$ varsayımı altında geçerlidir⁵⁴.

2.9.2. Tek Kanallı Tam Öncelikli Olmayan Modeller İçin Sistem Performans Ölçüleri

$$W_k = \frac{a_k}{b_{k-1}b_k} + \frac{1}{\mu_k}, \quad k = 1, 2, \dots, N$$

$$a_k = \sum_{i=1}^k \frac{\lambda_i}{\mu_i^2}$$

$$b_0 = 1$$

$$b_k = 1 - \sum_{i=1}^k \frac{\lambda_i}{\mu_i}$$

Model, $\sum_{i=1}^k \frac{\lambda_i}{\mu_i} < 1$ varsayımı altında geçerlidir⁵⁵.

2.9.3. Tam Öncelikli Modeller İçin Sistem Performans Ölçüleri

Tam öncelikli modeller için ortalama servis sürelerinin aynı olduğu varsayımının yapılması gerekmektedir. Tam öncelikli olmayan modeller için

⁵⁴ Hillier ve Lieberman, s: 881

⁵⁵ Hillier ve Lieberman, s: 882

notasyonlar aynı olmakla beraber k sınıfı önceliği için sistemdeki toplam bekleme süresiyle ilgili şu düzeltmenin yapılması gerekmektedir:

$$W_k = \frac{1/\mu}{B_{k-1}B_k}, \quad k=1,2,\dots,N$$

$$L_k = \lambda_k W_k, \quad k=1,2,\dots,N$$

Tek Kanallı Model için de $c=1$ alınarak hesaplama yapılır. Üssel dağılımın özelliğinden dolayı sistemdeki diğer müşteriler öncelikten etkilenmeyeceklerdir. Diğer müşteriler için ortalama servis süresi süreç boyunca $1/\mu$ 'dür⁵⁶.

⁵⁶ Hillier ve Lieberman, s: 883

3. SİMÜLASYON YÖNTEMİ

3.1. Simülasyonun Tanımı ve Yöntemin Tarihçesi

Gerçek hayatta karşılaştığımız problemlerin matematiksel veya istatistiksel olarak çözümünü çoğu zaman kolay olmayabilir. Bu tarz durumlarda problemin çözümü için gerçeğe en yakın şekilde modellemek ve varsayımları aksatmayacak çözüm önerileri getirmek gerekmektedir. Simülasyon tekniği de bu amaçla kullanılır. Ayrıca gerçeğin teoriden farklı olması ve herhangi bir araştırma üzerinde (Örneğin kuyruk modelleri) sistemi tanımlayacak genel bir modelin oluşturulamamış olması çoğu zaman simülasyon tekniğini zorunlu kılmaktadır.

Simülasyon kelimesinin tam olarak dilimizdeki karşılığını bulmak zor olabilir. Dilimizde benzetmek, taklit etmek gibi anlamlarda kullanılsa da bu konu üzerinde yapılan spesifik çalışmalarda olduğu gibi, tam karşılığı olacak şekilde; yani simülasyon kelimesi olarak kullanılması daha doğru olur.

Genel ve daha gerçekçi bir tanımını yapmak istersek, simülasyon; bir işletmenin ya da ekonomik sistemin davranışını belirten matematiksel ve mantıksal modeller ile bilgisayarda deneyler yapmak için kullanılan sayısal bir tekniktir; veya gerçeklere ve varsayımlara dayalı olarak belirsizlik koşulları altında seçenekleri değerlendirmek için gerçek karar vermeyi temsil eden ve bilgisayara programlanmış matematiksel bir model kullanan kantitatif bir tekniktir⁵⁴.

1777 yılında G. Comte de Buffon'un bir integral probleminin çözümü için yaptığı denemeler rassal örneklemenin ilk kullanımı olarak kabul edilir. Bu denemede Buffon, yatay bir düzlem üzerine d aralıklarla çizgiler çizerek L boyutundaki bir iğneyi bu düzlem üzerine gelişigüzel bırakmış, bu iğnenin düzlem üzerindeki doğrulardan herhangi biriyle kesişme olasılığını hesaplamaya çalışmıştır. P olasılığını hesaplamak

⁵⁴ Sariaslan, s: 37

için Buffon bu deneyi çok kez tekrar etmiştir. Deneyinin sonucunda Buffon bu olasılığı; $p = 2L/\pi d$ olarak hesaplamıştır⁵⁵.

Bu deneyden bir süre sonra Laplace, toplu iğne deneyinin çok sayıda tekrarlanmasıyla π sayısının hesaplanabileceği fikrini ortaya atmıştır. Lord Kelvin, daha sonra gazların kinetik enerjisi teorisindeki kinetik enerji kısmını integral denklemleriyle hesaplamak için tesadüfi örneklemeden yararlanmıştır. W. S. Gosset de korelasyon katsayısının dağılımını hesaplayabilmek için tesadüfi örneklemeden faydalanmıştır⁵⁶.

Günümüzde bilimsel disiplinlerde çok sık kullanılan bir kavram olan simülasyon terimi, çok eski çağlardan beri bilinen model kurma işlemini ifade eder, En geniş kapsamı ile simülasyon; gerçeğin sözel, şekilsel ya da simgesel olarak temsil edilmesidir.

Günümüzde işletmecilik alanında yeni bir gelişme olan simülasyon; heykel ve resim yapımından yıldız savaşlarına, baraj planlamasından süpersonik uçak modellerine kadar çok değişik alanlarda kullanılan bir kavramdır. Bu gelişim süreci ile simülasyon, bilgisayar kullanımını gerekli kılan matematiksel bir model aracılığı ile gerçek bir sistemin temsil edilmesini sağlayan bir tekniktir⁵⁷.

3.2. Simülasyonun Yöneylem Araştırmasındaki Yeri Ve Önemi

Simülasyon, yöneylem araştırmasında kullanılan çok sayıdaki teknikten bir tanesidir. Aynı zamanda diğer yöntemlere nazaran daha az kuralcı olan ve karmaşık tekniklerin yerine basit ve şematik olarak çözüm önerileri getiren bir tekniktir. Bir diğer üstünlüğü de hemen hemen bütün teknikler üzerinde uygulama ve deneme yapma şansının bulunmasıdır. Örneğin çoğu zaman çözümü zor ve karmaşık hale gelen kuyruk modellerinde, envanter yönetiminde, ulaştırma problemlerinde ve daha bir çok yöntem üzerinde uygulanabilirliği vardır.

⁵⁵ Aybaba Haçerlioğulları, **Monte Carlo Simülasyon Metodu Ve MCNP Kod Sistemi**, Kastamonu Eğitim Dergisi, Cilt No: 14, Sayı: 2, Ekim 2006, s:553

⁵⁶ Malvin H. Kalos, Paula A. Whitlock, **Monte Carlo Methods**, 2. Basım, Almanya, Wiley – VCH, 2004, s: 4

⁵⁷ Fehmi Dinçer, **Proje Değerlendirme Sürecinde Çağdaş Bir Yaklaşım: Monte Carlo Simülasyonu**, 22 Ocak 2007, <http://fehmidincer.blogspot.com/2007/01/nide-ili-tarihine-bir-baki-fehmi-dincer.html>, s: 1

Yöneylem araştırmasında kullanılan yöntemlerin üstünlükleriyle ilgili ORSA'nın (Operations Research Society of America) yaptığı bir araştırma dikkat çekicidir. 250 kişilik eğitimci ve 250 kişilik uzmandan oluşan rassal denekler üzerinde ve 1973, 1978 ve 1983 yıllarında yapılan bu çalışmada yöneylem araştırmasının tekniklerini önem derecelerine göre sıralamaları istenmiş ve aşağıdaki bulgulara ulaşılmıştır.

Tablo 6
Yöneylem Araştırmasında Kullanılan Yöntemlerin Üstünlükleriyle
İlgili Yapılmış Bir Araştırma

Teknik	1973		1978		1983	
	Eğitimci	Uzman	Eğitimci	Uzman	Eğitimci	Uzman
	N=106	N=92	N=98	N=88	N=89	N=96
	Sıra	Sıra	Sıra	Sıra	Sıra	Sıra
İstatistik	1	1	2	1	3	1
Lineer Programlama	2	2	1	2	1	3
Simülasyon	3	4	3	4	2	2
Matematiksel Programlama	4	7	10	5*	6	5
Olasılık	5	3	9	3	7	4
Stokastik Süreçler	6	-	4	7*	4*	8
Network Analizi	7	-	6	11*	4*	-
Optimizasyon	8	9*	-	5*	9	6
Dinamik Programlama	9	-	-	-	10	-
Karar Teorisi	10*	9*	5	-	8	-
Kuyruk Modelleri	10*	5*	8	7*	11	7
Envanter Yönetimi	10*	-	7	-	-	-

Kaynak: Thomas J. Schriber, An Introduction To Simulation Using GPSS/H,1991, sf:5)

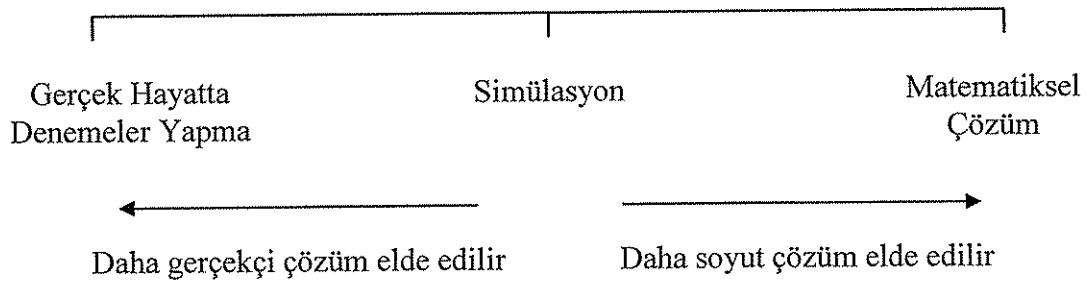
* İşareti ondalık rakamların bulunduğunu göstermektedir

- İşareti ilk 12'ye giremeyen teknikleri göstermektedir.

1973 yılında 106 eğitimci bu soruya cevap vermiş ve simülasyon 12 teknik arasında 3. sırada yer bulabilmiştir. Aynı deneme 1978 yılında yapıldığında 250 eğitimciden 98'i soruyu yanıtlamış ve simülasyon yine önem derecesine göre 3. sırada yer almıştır. 1983 yılında ise 250 eğitimciden 89'u soruyu yanıtlamış ve simülasyon 2. sıraya yükselmiştir. Aynı sorular yine rassal olarak seçilen 250 kişilik uzmanlardan

oluşan deney grubuna sorulmuş ve benzer şekilde 1973 yılında 4. sırada bulunan simülasyonun 1983 yılında 2. sıraya yükseldiği görülmüştür.

Simülasyonun diğer yöntemlerle ilişkisini ortaya koyan daha kesin bir örnekle durumu açıklayabiliriz. Aşağıdaki şekilde herhangi bir gerçek probleme çözüm önerisi olarak kullanılan üç yöntem bulunmaktadır⁵⁸.



Şekil 5. Alternatif Problem Çözüm Yöntemleri

Yukarıdaki şekli basit bir örnek üzerinde açıklayabiliriz. Sabit bir uzaklıktan bir hedefe ok atıldığını düşünelim. Atılan okun hedefi vurma olasılığını hesaplamak için gerçek hayatta denemeler yapma yöntemini kullanalım. İhtiyaç duyulan nesneler belli sayıda ok, hedef tahtası ve yaydır. Atılan okun hedefi vurma veya vurmama olasılıklarının hilesiz bir demir para deneyinde yazı veya tura gelme olasılıklarına eşit olduğunu düşünelim. Dolayısıyla okun hedefi vurma veya ıskalama olasılıkları 0,5'tir. Hedefe 10 ok atıldığını ve başarılı vuruşların sayısını kaydettiğimizi düşünelim. Ancak gerçek başarı yüzdesini hesaplamak için çok sayıda deneme yapılması gerekmektedir. Bu olay Ulam'ın 52 adet oyun kağıdıyla yaptığı denemeyle hemen hemen aynıdır. Deneme sayısının artması hem sıkıcı bir aktivite olacak hem de zaman kaybına yol açacaktır.

Başarı şansını hesaplamanın bir diğer yolu da olasılık teorisinden faydalanmaktır. Bernoulli dağılımı yardımıyla belli sayıdaki denemede okun hedefi

⁵⁸ Thomas J. Schriber, *An Introduction To Simulation Using GPSS/H*, New York – ABD, Wiley, 1991, s:5

vurma veya vurmama olasılığını tam ve kesin olarak hesaplayabiliriz. Hedefe 10 atış yapıldığını düşünürsek P okun hedefi vurma olasılığını

$$P = \sum_{k=0}^5 \binom{10}{2k} p^{2k} (1-p)^{10-2k}$$

olarak hesaplayabiliriz⁵⁹.

Bu probleme üçüncü çözüm yolu olarak simülasyon yöntemini önerebiliriz. Simülasyon, çözüme ne gerçek hayattaki deneme ne de matematiksel modelleme yöntemi ile yaklaşır. Simülasyon, gerçek sistemin bir alternatifi olarak düşünülebilir. Yukarıdaki örnekte hedefe gerçekten ok ve yay kullanarak atışlar yapıp zaman kaybetmek yerine aynı sonucu veren hilesiz madeni paranın atıldığını düşünelim. Yazı başarılı vuruşları, tura başarısız vuruşları simgelesin. Bu deneme de bir süre sonra sıkıcı olmaya başlayacaktır. Bunun yerine bilgisayarda simülasyonu uygulamak için 0-1 aralığında ve düzgün dağılmış bir dağılım kullanılabilir. Bilgisayarın tuşuna her basıldığında 0-1 aralığında rassal olarak bir sayının ekrana geldiğini düşünelim. 0,5 ve daha düşük sayılar başarılı vuruşları, 0,5'ten daha büyük sayılar da başarısız vuruşları simgelesin (Örneğin 0,3458 başarılı bir vuruş, 0,6507 başarısız bir vuruş olsun). Bu deney gerçek bir okla veya hilesiz bir parayla yapılan denemeden hem daha kısıda sonuca götürecek hem de daha zahmetsiz olup zaman kaybını önleyecektir.

Gerçek hayatta problemler okla yapılan başarılı vuruşların oranını hesaplamaktan çok daha ciddi ve matematiksel olarak modellenemeyecek kadar karmaşık olabilir. Bu gibi durumlarda simülasyon tekniği tek çözüm yöntemi olarak karşımıza çıkar⁶⁰.

3.3. Simülasyon Tekniğinin Kullanıldığı Alanlar

Simülasyon sosyal bilimlerin gelişmesiyle günümüzde hemen hemen her alanda uygulama şansı bulmaktadır. Bu yüzden herhangi bir kullanım kısıtlamasının yapılması doğru değildir. Fakat daha çok istatistiksel ve olasılıksal hesaplamaların

⁵⁹ Schriber, s: 6

⁶⁰ Schriber, s: 6

gerekli veya mümkün olduđu durumlarda kullanılır. Özellikle Monte Carlo simülasyonu olasılık hesaplarına dayandığından çođu sosyal bilimcinin çalışmalarını deney ortamında ve laboratuvarlarda yapabilme şansı doğmuştur. Ayrıca bu konu simülasyon yönteminin avantajları arasında da sayılabilir.

Simülasyon, sistem performansının bir çok faktöre bağımlı olarak nasıl değiştiğinin gözlemlenmesine olanak sağlar. Simülasyonu kullanarak sistem içerisinde birbirlerine bağı olan çalışanların iş gücü, süreç hata oranları, aktivitelerin gerçekleşme süreleri, parçaların sisteme geliş ve ilerleme hızları, maliyet ve gelirler, üretim hedefleri, personel ve makine kullanım yüzdeleri gibi değişkenleri, kolayca inceleyip süreç iyileştirilmesi yapılabilir. Bunun yanında alternatif senaryolar geliştirilerek yeni yatırım yapılabilecek alanları belirlemek ve risk faktörünü de göz önünde bulundurarak önerilen bir sistem üzerinde fizibilite çalışması yapmak da uygulamanın diğeri bir boyutu olmaktadır⁶¹.

Simülasyonun kullanıldığı alanları aşağıdaki tablo üzerinden daha iyi anlayabiliriz.

⁶¹ Yrd. Doç. Dr. Tuncer Özdil ve Prof Dr. Cengiz Yılmaz, İstanbul Menkul Kıymetler Borsasında Sektör Bazında İşlem Gören Hisse Senetlerinin Alım Satım Kararlarında En Yüksek Getirili Stratejinin Belirlenmesi, **Yönetim Ve Ekonomi**, Cilt: 13, Sayı: 2, 2006, s: 4

Tablo 7

Simülasyonun Kullanıldığı Alanlar

Şehir Yaşantısı ve Sosyal Hayatla İlgili Kullanımı: Karışık trafik sistemlerinin dizaynında ve çözümünde Çöp toplama ve depolama sorununun çözümünde Ambulans, itfaiye ve polis araçlarının acil ulaşım sorununun çözümünde Trafik ışıklarının sıralanmasında ve zamanlamasında Polis devriye bölgesi dizaynında Politik kampanya stratejilerinde Hava kirliliğinin kontrolünde Nüfus planlamasında Hava alanlarının dizaynında
Sağlık Sisteminde Kullanımı: Acil servis dizaynında Organ nakli stratejilerinde Hastane çalışanlarının mesai planlamasında Hasta trafiğinin kontrolünde Hasta kabul sisteminde Büyük ve yaşamsal önemi olan (Felaket sayılabilecek) hastalıkların kontrolünde Kan bankalarının yönetiminde Ambulans çalışanlarının mesai planlamasında
Havacılık ve Askeri Alanlarda Kullanımı: Uzay araştırmalarının güvenilirliğinin belirlenmesinde Savaş oyunlarında veya stratejilerinde Askeri birliklerin acemi er seçimlerinde Arama - kurtarma stratejilerinde Uzay savunma sistemlerinde Araştırma uydularının konumlanmasında
Hizmet Sektöründe Kullanımı: Portföy yönetiminde Sigortacılık ve risk işletmelerinde Profesyonel sporcu seçme stratejilerinde Komünikasyon ağı dizaynında Gemi filolarının çalışma programlamasında Süpermarket kasiyerlerinin mesai programlamasında Telefon sistemlerinde
Endüstri Alanında Kullanımı: Yiyeceklerdeki kimyasal bileşenlerin oranının çözümü problemlerinde Envanter yönetiminde Tamirci personelin çalışma saati programlamasında Dağıtım kanallarının dizaynında Ürün güvenilirlik testlerinde Kalite kontrolünde

Kaynak: Schriber, s:4

3.4. Simülasyonla İlgili Tanımlar

Simülasyon sürecinin ilk aşaması sistemin tanımlanmasıdır. Sistemin içereceği kısıtlar belirlenerek gerçeği temsil edecek bir model kurulur. Daha sonra da bu model üzerinde denemeler yapmak üzere bir bilgisayar programı oluşturulur veya mevcut programlardan faydalanılır. Elde edilen verilerele bu program üzerinde denemeler yapılır ve modelin gerçeği ne şekilde temsil ettiği saptanır. Deneme süreci pek çok kez tekrarlanarak gerçeğe yakın bulgular elde edilmeye çalışılır.

Simülasyon çıktısının tahmini tesadüfi örnekleme temeline dayanır. Gerçek durum gözlemlendiği zaman da aynı şey yapılır. Bu ifade simülasyon çıktısının rasgele değişimlere maruz kaldığı anlamına gelmektedir. Bu yüzden de herhangi bir istatistiksel deneyde olduğu gibi simülasyonda da istatistiksel yorum testlerinin kullanılması gereklidir⁶².

Öncelikle sistemin tanımının yapılmasında fayda vardır. Sistem; ortak bir amacı gerçekleştirmek üzere bir araya gelen, birbirlerine bağlı olarak çalışan ve birbirlerini etkileyen elemanlardan oluşan topluluktur. Sistemin çevresi ise sisteme ait olmayan fakat sistemi etkileyebilen elemanlardan oluşur. Örneğin bir mimar evi dizayn ederken betonarme yapıyı, ısıtma sistemini, elektrik sistemini, su sistemini, aydınlatma sistemini ve evle ilgili olan her türlü ayrıntıyı düşünmek zorundadır. Bir elektrik mühendisi ise bu evin bir bileşeni olan elektrik sistemiyle ilgilenir. Diğer bileşenler elektrik sisteminin çevresini oluşturur. Görüldüğü gibi evin kendisi bir sistem olarak tanımlanırken aynı zamanda alt sistemlerden oluşmuştur.

Her sistemin bir çevreye sahip olması gerekmez. Böyle bir durumda sistem tamamıyla içine dönüktür, sistemin bileşenleri dış güçlerden etkilenmez.

Sistemin bileşenlerinden herhangi birinde meydana gelen bir değişiklik olay olarak adlandırılır. Sistemde herhangi bir zaman aralığında bir değişiklik olmuyorsa sistem statiktir. Aksi halde sistem dinamiktir denir; sistemde olay veya olaylar gerçekleşmektedir.

⁶² Taha, s: 666

Bir sistem belirlenirken, sistemin karakteristiklerini en iyi şekilde ortaya koyacak değişkenlerden faydalanılmalıdır. Çok fazla değişken kullanmanın modeli o kadar iyi temsil edeceği söylenemez. Tam tersine bu hem içinden çıkılamayacak kadar karmaşık bir durum oluşturur hem de maliyet ve zaman kaybına neden olur. Çok fazla sadeleştirmenin yapılması da modeli tam olarak tanımlayamayabileceği gibi arzu edilmeyen hatalı sonuçların ortaya çıkmasına da sebep olabilir.

Durum, sistemin herhangi bir anında veya herhangi bir aralığında bileşenlerin özelliklerinde meydana gelen değişiklikler olarak adlandırılabilir. Durum değişkeni göz önünde bulundurulursa sistem kesikli ve sürekli sistemler olarak ikiye ayrılır. Kesikli sistem zamanın herhangi bir t anında gözlenen değişiklikleri veya olayları içerir. Örneğin kuyruk sistemine dahil olan veya sistemde hizmet görüp ayrılan müşteriler kesikli sistemin elemanlarıdır. Sürekli sistemlerse herhangi bir t anında gözlenmesi bir şey ifade etmeyen, daha çok belli bir zaman aralığındaki değişikliklerle tanımlanabilen sistemlerdir. Bir reaktörde gözlemlenen kimyasal değişimler veya reaktörden belli bir süre boyunca geçen kimyasal bileşenlerin yol açtığı değişiklikler sürekli sisteme örnek verilebilir.

Uygulamada tam anlamıyla kesikli veya tam anlamıyla sürekli bir sistemle karşılaşma olasılığı oldukça düşüktür. Genellikle sistemler bu iki sistemin özelliklerini taşır niteliktedir. Bu durumda hangi sistemin özellikleri ağır basıyorsa sistem o grupta değerlendirilebilir⁶³.

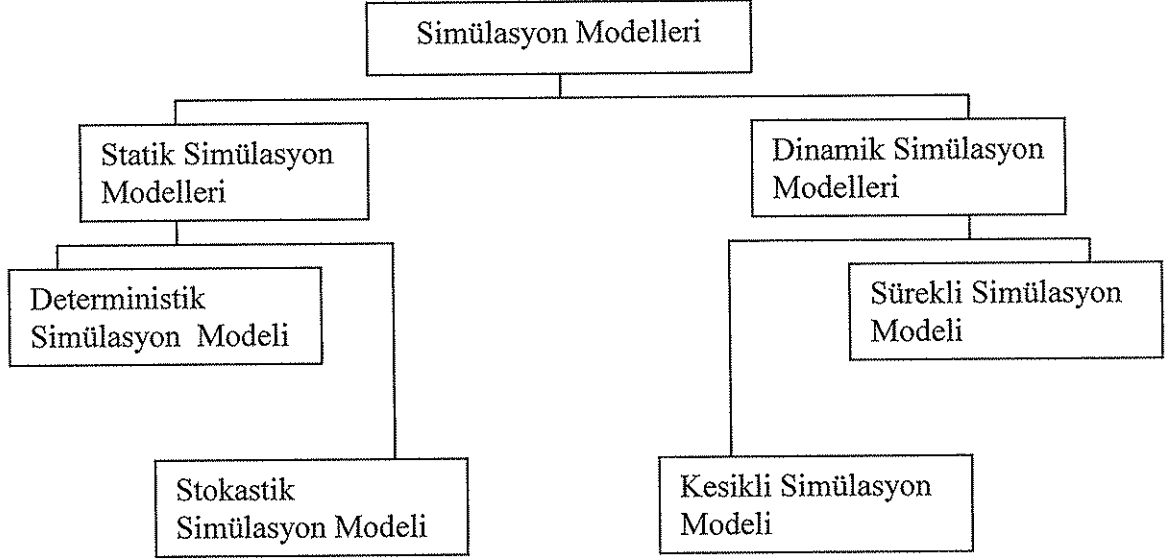
Model, yapılan araştırmanın özelliklerine uygun şekilde elemanların ve bunlar arasındaki ilişkilerin tanımlanmasıdır. Simülasyon amacıyla kurulan modele simülasyon modeli denir. Simülasyon modeli genellikle sistemin işleyişiyle ilgili varsayımlar doğrultusunda sistemin önemli bileşenleri arasındaki ilişkileri matematiksel veya mantıksal sembollerle açıklayan bir kümedir⁶⁴.

⁶³ Nalan Cinemre, **Yöneylem Araştırması**, 2. Basım, İstanbul, Beta Basım Yayın Dağıtım A. Ş., 2004, s: 642

⁶⁴ Cinemre, s: 642

3.5. Simülasyon Modelleri

Simülasyon modellerini kesin olarak birbirinden ayırmak zor bir işlem olabilir; fakat uygulamada kolaylık olması açısından aşağıdaki şekilde sınıflandırılması yapılabilir:



Şekil 6. Simülasyon Modellerinin Sınıflandırılması

3.5.1. Statik Simülasyon Modeli

Zamanın herhangi bir t anında sistemin durumunu gösteren modellerdir. Zaman sistem üzerinde etkili değilse statik simülasyon modelleri kurulabilir. Bu tarz modeller genellikle Monte Carlo simülasyon modeli olarak adlandırılır.

3.5.2. Deterministik Simülasyon Modeli

Olasılıksal değişken içermeyen simülasyon modelleridir. Örneğin fiziksel deneyler deterministik modeller sınıfına girer. Suyun normal şartlar altında kaynaması deneyi veya karı girdi ve çıktı arasındaki fark olarak tanımlayan modeller deterministik modellerdir.

3.5.3. Stokastik Simülasyon Modeli

En az bir tane rassal değişken içeren simülasyon modelleridir. Deneme sonuçları bu değişkenin alacağı değere göre farklılık gösterir.

3.5.4. Dinamik Simülasyon Modeli

Sistemin zaman boyutundaki gelişmesini gösteren modellerdir. Modeldeki değişkenin kesikli veya sürekli olması halinde kurulacak olan simülasyon modeli de Kesikli veya Sürekli Simülasyon Modeli olarak adlandırılacaktır.

3.6. Simülasyon Modellerinde Kullanılan Gözlemler

Simülasyon istatistiksel bir deneydir. Bu deneyin sonuçları özel istatistiksel yorumlama araçları (Güven aralıkları, hipotez testleri,...) yardımıyla yorumlanmalıdır. Bunun için de bir simülasyon deneyindeki gözlemler şu üç koşulu sağlamalıdır:

- 1) Gözlemler hareketsiz, belli dağılımlardan elde edilmiş olmalıdır.
- 2) Gözlemler, normal anakütleden alınmış örneklem olmalı.
- 3) Gözlemler bağımsız olmalıdır.

Kurallara çok bağlı olarak düşünüldüğünde, bu koşulların hiçbiri simülasyonda tam olarak gerçekleştirilemez. Bununla birlikte simülasyon gözlemlerinden sonuçlar çıkararak kısıtlı da olsa bu koşulların istatistiksel olarak devamlılığı sağlanabilir⁶⁵.

Örneklem hacmi büyüdükçe sonuç kesin bir değer etrafında kararlı olmaya başlar. Dolayısıyla bu kararsızlık etkisi yeterince büyük örneklem hacimleri kullanmak suretiyle etkisiz hale getirilebilir⁶⁶.

Simülasyonun sonucu, simüle edilen periyot uzunluğunun fonksiyonudur. İstikrarsız davranışların ilk periyodu genellikle geçiş ya da alışma periyodu adını alır. Sonuçlar kararlı hale geldiği zaman sistemin kararlı halde davrandığı söylenir. Aradaki

⁶⁵ Taha, s: 692

⁶⁶ Taha, s: 692

geçiş periyodunun uzunluğu modelin karakteristiklerinin fazlalığına bağlı olup kararlı halin ileride hangi noktada başlayacağını kestirmeye yarayan bir yöntem mevcut değildir. Genelde daha uzun simülasyon koşumu daha iyi bir kararlı hale ulaşma şansına sahiptir.

İkinci koşul Merkezi Limit Teoremi kullanılarak sağlanır. Merkezi Limit Teoremi, örneklemin alındığı anakütleye bakmaksızın bir örneklem ortalaması dağılımının asimptotik normal olduğunu söyler.

Üçüncü koşul ise gözlemlerin bağımsızlığıyla ilgilidir. Simülasyon deneyinin yapısı birbiri peşisıra yapılan simülasyon gözlemleri arasında bağımsızlık olmasını garanti edemez. Bununla birlikte simülasyon gözlemlerini temsil etmek üzere örneklem ortalamalarının kullanılması bu sorunu hafifletebilir. Bu durum, özellikle örneklem ortalamasını hesaplamada kullanılan zaman arttığında doğru olmaktadır⁶⁷.

Verilen bir rasgele sayı dizisi ile yapılan bir deneyde sistemin çalışması gözlenir. Bunun için bir veya birkaç ölçü değişkeni (Örneğin atölyede herhangi bir anda çalışır durumda olan makinelerin sayısı) gözlenir ve bu gözlemlerin zamanla değişmesi incelenir. Deney, başka bir rasgele sayı dizisi ile tekrarlanırsa farklı bir netice elde edileceği açıktır. Sistemin büyüklüğü ve iç bağlantıları, modelde temsil edilebilen ayrıntılar, kullanılan dağılım fonksiyonları istatistiksel kararlılığa etkisi olan faktörlerdir. Gerçek sistem üzerinde de dış faktörlerin rasgele etkiler yapacağı farz edilirse bir deney sonucunun elde edilmesi mümkün olan sonuçlardan biri olduğu görülebilir. Monte Carlo simülasyonu metodunun temel ilkesi budur⁶⁸.

3.7. Simülasyonun Kullanım Amaçları

Daha önce simülasyonun kullanımı ile ilgili herhangi bir sınırlamanın yapılamayacağından ve hemen hemen her bilim dalında kullanıldığından bahsetmiştik. Fakat genel olarak bir kullanım kalıbı belirlemek istersek aşağıdaki durumlarda kullanabileceğimizi söyleyebiliriz;

⁶⁷ Taha, s: 692

⁶⁸ M. Nadir Yücel, **Monte Karlo Metodu**, İstanbul, İstanbul Teknik Üniversitesi Elektronik Hesap Bilimleri Enstitüsü Yayınları:14, 1973, s: 62

1) Simülasyon, karmaşık bir sistemin iç yapısını veya karmaşık bir sistemdeki alt sistemleri incelemek için kullanılabilir.

2) Bilgi, organizasyonel ve çevresel değişiklikler simüle edilebilir ve modelin davranışı üzerinde bu değişikliklerin etkileri incelenebilir.

3) Bir simülasyon modelinin tasarımından elde edilen bilgiler, incelenen sistemin geliştirilmesine büyük ölçüde katkıda bulunmaktadır.

4) Simülasyon girdilerini değiştirerek ve sonuçları inceleyerek, hangi değişkenlerin daha önemli olduğu ve değişkenlerin birbirlerini nasıl etkiledikleri hakkında bilgi edinilir.

5) Simülasyon, analitik çözüm metodolojisini destekleyen bir bilgi verici araç olarak kullanılabilir.

6) Simülasyon, uygulamadan önce yeni tasarımlar ve politikalar deneyerek durumun ne olacağını görmek için kullanılabilir.

7) Simülasyon, analitik sonuçları test etmek için kullanılabilir.

Bunun yanında simülasyona gerek duyulan ve simülasyonun kullanılabileceği alanları şu şekilde sıralayabiliriz;

1) Sistemin davranışlarını inceleme ve tanımlama,

2) Gözlenen sistem davranışlarını açıklayan teori ya da hipotezleri deneme ve uygulama,

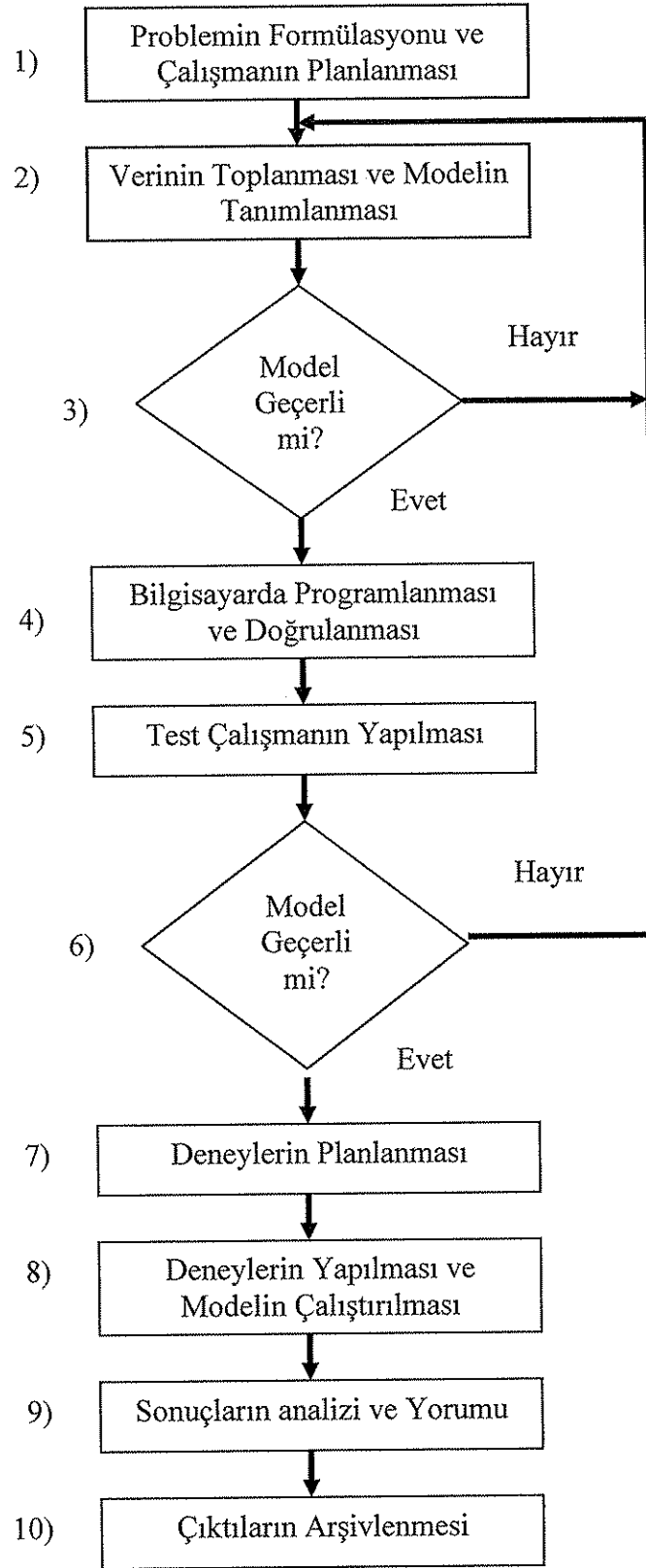
3) Bu teorileri kullanarak sistemdeki değişmelerin etkilerini belirleme ve böylece sistemin gelecekteki davranışını tahmin etme⁶⁹.

⁶⁹ Sarıaslan, s:44

3.8. Simölasyon Sürecinde İzlenmesi Gereken Adımlar

Bir simölasyon çalışması pek çok farklı şekilde tanımlanabilir. Çalışmalarda en çok kullanılanı 1986 yılında Law'ın geliştirdiğı aşağıdaki şematik çizelgedir⁷⁰ ;

⁷⁰ Schriber, s: 11



Şekil 7. Simülasyon Çalışmasında İzlenen Adımlar

3.8.1. Problemin Formülasyonu ve Çalışmanın Planlanması

Çalışmanın ilk aşamasıdır. Bu aşamada modeli kuracak olan kişi ve müşteri bir araya gelerek her türlü ayrıntıyı gözden geçirir. Modeli kuracak olan kişi ile müşteri aynı kişi de olabilir. Bu durumda modeli kuran kişinin incelenen konu hakkında bilgi sahibi olması gerekmektedir. Modelin sağlıklı oluşturulabilmesi için her türlü alternatif düşünülmelidir. Farklı alternatifler arasından seçim yaparken zaman ve maliyet kısıtları önemli yer tutar⁷¹.

Çalışmanın planlanması aşamasında modeli kuracak olan kişi çalışmanın kavramsal bir iskeletini oluşturur. Modelde incelenecek ana olaylar ve elemanlar incelenir. Şematik olarak yapılan çalışma ve grafiksel gösterimler tüm detayların belirlenmesinde ve sistemdeki akışın ve etkileşimlerin detaylı olarak gözlenmesinde yardımcı olur⁷².

3.8.2. Verilerin Toplanması Ve Modelin Tanımlanması

Simülasyon çalışmasının yapılması için veri setine ihtiyaç duyulur. Bunun için de gerçekte varolan bir sistem çalışırken bu sistem üzerinde gözlem yapılması ve verilerin toplanması önemlidir. Bu veri; üretim sürecinde bir malın ortalama üretim süresi, makinenin parçalarının ortalama değişme süresi, herhangi bir parçası değişen bir makinenin bir sonraki arızaya kadar geçireceği süre ve arızanın ortalama tamir süresi olabilir. Gerçekte varolan bu sistem incelenirken gelecekte yapılacak çalışma için veri elde edilmiş olur. Modellenmesi istenen sistem gerçekte varolmayan veya eksik verilerin bulunduğu bir sistem de olabilir. Böyle bir durumda sisteme en hakim kişilerden yardım alınabilir, modeli kuracak olan kişi veriyi kendisi toplayabilir veya verilerle ilgili tahmin yapabilir.

Bu aşamada oluşturulan model sistemin mantıksal bir halidir. Her adımda yeni bilgiler eklenerek çalışmanın son aşamasına kadar geliştirilir. Soyut olan bu model değişkenler arasındaki ilişkileri tanımlar.

⁷¹ Schriber, s: 11

⁷² Simülasyon Ve Modelleme Nedir?, 10 Ekim2006, <http://enm.blogcu.com/1179385/>, (21 Ocak 2008), s:7

Tüm bunlara ek olarak bir simülasyon modeli için;

- 1) Parametreleri tahmin etmek için verilere ihtiyaç vardır.
- 2) Modelin başlangıç değerlerinin belirlenmesinde çoğu kez veri gereklidir.
- 3) Geliştirilen modelin gerçeği temsil edip etmediği gerçeğe ait veriler aracılığıyla değerlendirilir.

Ancak Monte Carlo modelleri yapay veri üretme olanağı verdiklerinden uzun dönemleri kapsayan verilerin derlenmesini gerektirmezler. Bu nedenle uzun dönemli verilerin bulunmadığı durumlarda araştırmacılar bu modelleri tercih ederler. Fakat yapay veri üretimine kaynaklık eden olasılık dağılımlarının belirlenmesinde yine sağlıklı verilere ihtiyaç vardır. Kuşkusuz modelin gerçeğe uygunluğunu değerlendirmede aynı ihtiyaç söz konusudur⁷³.

3.8.3. Modelin Geçerliliğinin Doğrulanması

Model, amaca uygun bir şekilde çalışıyorsa ve sistemi temsil gücü yüksekse geçerliliğini doğrulayabiliriz. Amaca uymayan ve sistemi yansıtmayan modeller için 2. adıma geri dönülür ve model amaca uygun olarak tanımlanır. Modelin doğrulamasını yaparken simülasyon çalıştırılıp etkileşimler ve işlemler gözlenir. Çoğu zaman kompleks yapısı olan modellerin doğrulanması için bu işlem birden çok defa tekrarlanır. Görülen hataların düzeltilmesi için modelin bir başka model kurucusu tarafından çalıştırılmasından veya daha önce kurulan modellerin sonuçlarıyla karşılaştırılmasından faydalanılabilir⁷⁴.

3.8.4. Modelin Bilgisayarda Programlanması ve Doğrulanması

Bu adımda, 2. adımda tanımlanan ve 3. adımda geçerliliği doğrulanan modelin bilgisayar ortamına aktarılması sağlanır. Bunun için spesifik bir program kullanılır. Bilgisayarda tanımlanan bu modelin ve kullanılan programın da geçerliliğinin

⁷³ Sariaslan, s: 53

⁷⁴ Simülasyon Ve Modelleme Nedir?, 10 Ekim 2006, <http://enm.blogcu.com/1179385/>, (21 Ocak 2008), s:8

saptanması gerekmektedir. 2. adımda kurulan ve geçerliliği saptanan bu modelin bilgisayar ortamında da doğrulandığına veya modelin bir veya birden fazla mantık hatası içerdiğine karar verilir.

Modelin doğrulanması yapılırken sistem üzerinde çok büyük etkisi olmayan ve etkisi çok uzun sürmeyen hatalar göz ardı edilebilir. Esas olarak ilgilenmemiz gereken kurduğumuz modelin amacını yansıtamayacak derecede yüksek düzeydeki mantık hatalarıdır⁷⁵.

Modelin doğrulanmasında yapay veriler kullanılabilir. Bunun sebebi modelin gerçekte bu verilerle çalıştırılmayacak olması ve sadece modelin mantıksal olarak işleyip işleyemeyeceği konusu üzerinde yoğunlaşmış olmasıdır. Uygulamanın pratik olması açısından doğrulama adımı veriler toplanmadan önce veya verilerin toplanması esnasında yapılabilir. Aslında model bize toplamamız gereken veriyi modelin kurulması aşamasında gösterir. Eğer zamanla ilgili herhangi bir sıkıntı yaşanmıyorsa 2. adımdaki verilerin toplanması bölümü çıkarılıp önce model kurulur, sonra modelin işlerliği açısından hangi verilerin toplanacağına karar verilir.

Modelin geçerliliği saptanırken ‘‘Doğru Modeli’ mi kurduk? – Doğru konseptte bir model mi oluşturduk?’, modelin doğruluğunu saptarken de ‘Modeli doğru bir şekilde mi oluşturduk? – Konsept modeli bilgisayar ortamına doğru bir şekilde mi aktardık?’ sorularına yanıt aranır⁷⁶.

3.8.5. Test Çalışmasının Yapılması

Test çalışmasının yapılmasının bazı olumlu yönleri vardır. Öncelikle deneme veya test çalışması modeli oluşturan kişinin simülasyon modeli konusundaki tecrübesini arttırır ve beklentilerinin çıktı üzerinde ne şekilde gerçekleştiğini veya hangi nedenlerle gerçekleşemediğini görmesini sağlar. 8. aşamada modelin gerçek simülasyonu yapılırken, simülasyonun farklı yönlerinin görülmesinde ve planlanmasında kullanılmak üzere bize bazı ipuçları ve sonuçlar sağlar. Gerçekte var olan bir sistemin geçerliliğinin sınanmasında kullanılmak üzere bize yardımcı olur.

⁷⁵ Schriber, s: 12

⁷⁶ Schriber, s: 13

3.8.6. Modelin Geçerliliğinin Doğrulanması

Bu aşamada model gerçekte var olan bir sistem için oluşturulmuşsa ve bu sistem için performans ölçüleriyle veriler toplanmışsa, modelin geçerliliğini modelden elde edilen bu performans değerleriyle gerçek sistemin performans değerlerini karşılayarak hesaplayabiliriz. Eğer modellenen sistemle gerçekte var olan sistemin davranışları örtüşüyorsa bir sonraki adıma geçilir; eğer arada büyük fark varsa modeli tekrar tanımlanması için 2. adıma geri dönlür.

3.8.7. Denemelerin Planlanması

Çoğu durumda projede yer alan kişilerin değişik alternatif çözümlere ilişkin temel ve basit fikirleri vardır. Her alternatifi model üzerinde denemeden önce kabul edilebilir sonuçlar elde etmek için gereken simülasyon sayısının ve modelin eğer mümkünse durağan duruma gelmesi için gerekli zamanın hesaplanması gerekmektedir. Durağan durum, simülasyon çıktılarının durağan veya düzgün olduğu anlamına gelmez; durağan durum, verilerin dağılımı ya da çıktılardaki istatistiksel büyüklüklerin zamanla değişmeyecek olmasıdır. Rassal karakteristikleri olan bir deneyde olduğu gibi simülasyon çalışmaları da rasgele olayları içerdiği için, simülasyon çıktıları da rasgele olacaktır. Bu sebeple tek bir simülasyon çıktısı oluşma ihtimali olan bir çok sonuçtan sadece bir tanesini yansıtır. Bu nedenle sonucu test etmek için bir çok defa denemenin tekrar edilmesi gerekmektedir⁷⁷.

Bu adımda; simülasyonun uygun çalışma süresi ne kadardır, kaç tekrar yapılmalıdır ve kaç farklı rassal değişken kaynağı kullanılmalıdır sorularına yanıt aranır.

3.8.8. Denemelerin Yapılması ve Modelin Çalıştırılması

7. adımda açıklanan işlemler bu adımda bilgisayar ortamında test edilir ve simülasyon çalıştırılır.

⁷⁷ Simülasyon Ve Modelleme Nedir?, 10 Ekim 2006, <http://enm.blogcu.com/1179385/>, (21 Ocak 2008), s:9

3.8.9. Sonuların Analizi ve Yorumu

Bu adımda 8. adımın sonuları istatistiksel olarak test edilerek sistem performans lleri tahmin edilir ve alternatif sistem dizaynları sıralanır.

3.8.10 ıktıların Dokmantasyonu

ıktıların dokmantasyonu, alıřmada izlenen adımların yazılı olarak raporlanmasına dayanır. alıřmada nemli grlen hususlar, ama, konsept modelin oluřturulmasındaki varsayımlar, veri toplama, konsept modelin bilgisayar ortamına aktarılırken izlenen yol, modelin geerliliğinin ve doğruluğunun saptanmasında izlenen yol, rnek ıktı, alıřmanın sonucu, alıřmayı gerekleřtiren řahıslar ve alıřmanın yapıldığı tarih yazılı bir řekilde dosyalanır. Bu iřlem zaman alıcı ve sıkıcı bir aktivite gibi grnse de daha sonra bu konuda yapılacak arařtırmalara ıřık tutacak ve bařka arařtırmacıların zaman ve maliyet kaybını ortadan kaldıracaktır.

3.9. Simlasyon Ynteminin Avantajları Ve Dezavantajları

Simlasyon tekniğinin, arařtırmacıya kazandırdığı avantajların yanında bazı dezavantajları da bulunmaktadır. Bu bulguların saptanması ve mevcut alternatifler zerinde karar vermek iin simlasyon alıřmasını yapacak kiři her trl ayrıntıyı dřnmek zorundadır.

Simlasyon yntemiyle daha nceki blmlerde verdiğimiz hedef tahtası, ok ve yayla yapılan deneme rneğinde olduėu gibi gereki sonulara ulařılmıştır; diğerk bir deyiřle gerek sistem zerinde yapılan deneme ile simlasyon tekniğıyle yapılan deneme bizi aynı sonuca gtrmüştür. Fakat gerek sistem zerinde deneme yapmamız iin bu sistemin deneme yapmadan nce varolması ve denemelerin bu sistem zerinde yapılması gerekmektedir. Aynı zamanda gerekte var olan ve dzgn bir řekilde alıřan sistemlerin zerinde bazı deneylerin yapılması da ok mantıklı olmayabilir. oėu kimyasal ve biyolojik deneyde olduėu gibi gerek denemeler insana zarar verebilir. Kimyasal silah geliřtirip etkisini denemek veya kansere karřı bir ila geliřtirip insan vcudundaki etkilerini gzlemlemek zararlı etkileri de beraberinde getirir. Bunun yanında sistemin ok karmařık olması, denemelerin ok fazla zaman alması ve ok

yüksek maliyete sebep olması da istenmeyen durumlar olarak araştırmacının karşısına çıkar.

Matematik ise çok güçlü bir yöntem olmasının yanında tam ve doğru sonuçlar verir. Kurulan denklemlerde değişkenlerden bir tanesi üzerinde meydana gelecek değişiklikle tüm model veya diğer değişkenler üzerindeki etki incelenebilir. En iyi sonucun bulunması ve bulunan çözümün en iyi çözüm olup olmadığının araştırması da matematiğin kullanılmasıyla yapılır. Matematiksel yöntemlerin de bu kadar üstün ve güçlü olmasına rağmen bazı dezavantajları bulunmaktadır. Soyutlamanın yüksek olması, bir veya birkaç karakteristiğin veya değişkenin dışlanmış veya yanlış seçilmiş olması matematiksel çözümün dezavantajlarıdır. İlk bakışta matematiksel çözüm doğruymuş gibi görülebilir; ancak değişkenlerin dışlanması, gözardı edilmesi veya gereksiz değişken kullanılması bizi yanlış sonuca götürür. Netice olarak araştırmacı; ‘yanlış problemin doğru çözümü, yanlış çözümdür’ ilkesinin doğruluğu olduğu yönünde bulguya ulaşır⁷⁸.

Herhangi bir problemle karşılaşıldığında akla öncelikle problemin analitik yaklaşımla çözülmesinin mümkün olup olmadığı sorusu gelir ve problemin analitik yaklaşımla çözülebilecek bir yapıya oturtulmasına çalışılır. Oysa analitik yaklaşımların hemen hepsi belirli varsayımların sağlanması durumunda uygulandıklarından , ulaşılan sonuçların geçerliliği ile sınırlı kalır. Yapılan varsayımlar geçerli olsalar bile, sırf analitik yöntemi mümkün kılmak için gerçeğin bir bölümünden vazgeçilmesi sorun yaratabilir. Bu gibi durumlarda daha az sayıda varsayım gerektiren ve varsayımların gerçekleşmemesi durumuna karşı daha az duyarlı olan simülasyon yolu ile çözüm uygun olur⁷⁹.

Kuşkusuz her kantitatif teknik gibi simülasyon tekniğinin de kendisine özgü bazı eksiklikleri ve sınırlılıkları olacaktır. Herşeyden önce simülasyon modelleri temelde deneysel nitelikli olduklarından istatistiksel deney hataları nedeniyle optimal sonucu garanti edemez. Ancak benzetim deneyleri doğru bir biçimde yapılırsa sonuçlar optimal çözüme çok yaklaşır. Diğer yandan güvenilir sonuçlara ulaşmak için deneylerin

⁷⁸ Schriber, s: 7

⁷⁹ Cinemre, s: 639

istenilen biçimde düzenlenmesi bilgisayar zamanı açısından simülasyon modellerinin maliyetini yükseltir⁸⁰. Ayrıca çoğu durumda matematiksel veya analitik yöntemlerle kolaylıkla çözülebilecek bir probleme simülasyon yöntemiyle çözüm aramak da istenmeyen sonuçlar doğurabilir. Çünkü tam ve kesin bir sonuca varmak istiyorsak matematiksel ve analitik yöntemlerin simülasyon yönteminden daha üstün olduğu açıktır.

Simülasyon yönteminin diğer avantajlı taraflarını şu şekilde sıralayabiliriz⁸¹;

- 1) Simülasyon esnek ve basit bir çözüm yöntemidir.
- 2) Simülasyon yönteminin uygulanması daha fazla aritmetik işlem gerektirmesine karşın, analitik işlemlerin uygulanmasından daha kolaydır.
- 3) Yalnızca analitik çözümlerin yerine değil analitik yaklaşımla belirlenen çözümlerin doğruluğunu gerçeklemek için de kullanılabilir.
- 4) Aşamalı olarak uygulayabilme imkanı vardır.
- 5) Klasik çözüm yöntemlerinin kullanılmadığı büyük karmaşık problemlerin çözümünde oldukça etkilidir.
- 6) Bir başka yöntemde incelenmesi olanaksız olan koşullar ve kısıtlar simülasyon ile rahatça modellenebilir.
- 7) Sonuçları ancak aylar, yıllar sonra alınabilecek durumlarda simülasyon ile çok kısa sürede analiz edilebilir.
- 8) Simülasyon, modellenen sistemi değiştirmeden yeni fikir ve politikaların model üzerinde rahatça uygulamasına olanak verir.
- 9) Kullanıcı simülasyonu istenen zamanda durdurup yeniden başlatabildiğinden deney koşullar üzerinde tam bir kontrole sahiptir.

⁸⁰ Sariaslan, s: 39

⁸¹ Haçerlioğulları, s:548

Bu kadar çok avantaja sahip olması araştırmacının, daha uygun olan analitik yöntemleri kullanmak yerine simülasyon yöntemi kullanması gibi bir hataya düşmesine sebep olabilir.

Yöntemin dezavantajları ise şunlardır⁸²;

- 1) Kesin olmayan bir yöntemdir ve pek çok durumda bu kesin olmayışlığı ölçmek çok zordur.
- 2) İyi bir simülasyon modelini geliştirmek vakit alıcı ve pahalıdır.
- 3) Simülasyon bir çözüm yöntemi değildir, sadece çözüm geliştirme yolu yaratır.
- 4) Optimum çözüm üretme garantisi yoktur. Bir çeşit deneme-yanılma yöntemidir.
- 5) Her simülasyon modeli kendine özgüdür.
- 6) Uygulamasındaki kolaylıklar dolayısıyla analitik çözümlerin göz ardı edilmesine neden olabilir.
- 7) Modelleme ve bulguların analizinde yapılacak hatalar, yanlış sonuçlara yol açabilir.

3.10. Simülasyon Türleri

Uygulamalarda sıklıkla kullanılan dört çeşit simülasyon modeli vardır. Bunlar⁸³;

3.10.1. Monte Carlo Simülasyonu

Olasılıksal süreçlerle ilgilidir ve sistemin ilişkileri rasgele değişkenlerle açıklanır.

⁸² Haçerlioğulları, s:548; Cinemre, s: 640

⁸³ Halaç, s: 444

3.10.2. Sistem Simülasyon Modelleri

Bazı işleyen sistemlerin karmaşık bir şekilde yeniden yaratılmasıdır. Monte Carlo simülasyonundan iki yönüyle farklıdır; rasgele sayılar yaratmak yerine anakütleden rasgele örnekleme yapılır, bir çözüme ulaşmak için matematik karar kuralları kullanılır. Ama matematik olarak karmaşıklık artarsa Monte Carlo simülasyonu kullanma olanağı doğabilir.

3.10.3. İlkesiz simülasyon Modeli

Yönetim kararlarının sözlü ifadesi şeklindedir. Karar problemlerinin pek çoğunda en iyi çözümü bulmak için hesaplama kuralları mevcut olmayabilir. Bunun sonucu olarak yönetim ilkesiz simülasyon kullanmak zorunda kalır. Bu yaklaşım, karar vermek için lüzumlu olan en az zamanda karar vermeyi sağlayan baş parmak kurallarını (Rules of Thumb) kapsar. Kantitatif analizler karar kurallarının geçerliliğini denemekte kullanıldığı için ilkesiz modeller matematikten arınmış değildir ama diğer simülasyon yaklaşımlarından daha sözlüdür.

3.10.4. Oyun Modelleri

Tarafları olan simülasyondur. Örnek olarak işletme problemlerini simüle etmek için karar verme ve sonuçlarını görmek amacıyla düzenlenmiş bilgisayar ile oyunlar olduğu kalem kağıtla yapılan çok sayıda oyun vardır. Bunlar öğrenci eğitim ve yönetici yetiştirme amacıyla kullanılırlar. Ayrıca dünya ekonomisi ile ilgili simülasyon oyunları, harp simülasyon oyunları, poker ve satranç simülasyon oyunları geliştirilmiştir.

3.11. Simülasyon Dilleri

Simülasyon dilleri ilk olarak 1960'lı yılların başlarında GPSS ve SIMSCRIPT'in takdimiyle ortaya çıkmaya başlamıştır. GPSS, kuyruk sistemlerinin modellenmesi amacıyla tasarlanmıştır, özel amaçlı bir simülasyon dilidir. Günümüzde, bu şekilde konuya göre tasarlanmış özel amaçlı bir çok simülasyon dili bulunmaktadır. SIMSCRIPT, model kurmada daha geniş tasarım olanakları sağlayan, genel amaçlı süreksiz olay simülasyon dillerini temsil etmektedir. Diğer genel amaçlı süreksiz olay simülasyon dillerine GASP IV, SLAM II ve SIMAN örnek olarak verilebilir.

Richard E. Nance isimli bir araştırmacı simülasyon dillerinin kabaca beş gelişim periyodundan geçtiğini kaydetmektedir. Bunlar;

1) 1955-1960 arası: Simülasyonlar özellikle FORTRAN olmak üzere, genel amaçlı programlama dilleri ile yapılmaktadır. Bu dönemde, simülasyon modellerinin çalıştırılabilmesi programlama dillerinin gelişmesine bağlı idi.

2) 1960-1965 arası: Simülasyon çalışmalarını gerçekleştirecek GPSS, SIMULA, SIMSCRIPT ve CSL gibi ilk simülasyon dilleri ortaya çıkmıştır. Bu diller alternatif yaklaşımlar sunarak bugün kullanılan birçok simülasyon dili için temel oluşturmuşlardır.

3) 1966-1970 arası: Simülasyon dillerinin ikinci jenerasyonu olarak adlandırılan bu dönemde GASP II, III, 360 ve V versiyonları geliştirilmiştir. Ayrıca SIMULA 67'nin yeni versiyonu kullanılmaya başlanmış, CSL yerini genişletilmiş hali olan ECSL'ye bırakmıştır.

4) 1971-1978 arası: Bu dönemde SLAM ve Q-gere gibi bazı yeni diller ortaya çıkmışsa da dönemin özelliği, mevcut dillere yeni özellikler eklenmesi olmuştur. Örneğin; SIMSCRIPT II.5'e sürekli modelleme yeteneği ve proses etkileşim stratejisi eklenmiştir. GASP IV, karma modeller tasarlayabilme yetkisini bünyesine dahil etmiştir. Bu dönemin sonlarına doğru bazı dillerin karşılıklı etkileşim versiyonları ortaya çıkmaya başlamıştır. Bu dönemde; Zeigler, modellemeye ait temel kavramları sunmuştur. Ören ve Zeigler, mevcut simülasyon dillerinin kritiklerini yaparak yeni dillerin içermesi gereken iki kavram ortaya koymuştur. İlk kavram; mantık olarak birbirinden farklı olan; modelleme, deney tasarımı ve çıktı analizleri safhalarının birbirinden ayrılmasıdır. İkinci kavram ise programın kanıtlanması ve modelin geçerliliği için veritabanı teknolojisinin, grafiklerin kullanılmasıdır.

5) 1979'dan günümüze kadar geçen süre : Bu dönemde programlamadan model geliştirmeye doğru kayan bir önem sırası olmuştur. Bu dönemde, program yazma paketler tarafından gerçekleştirilmeye başlanmıştır. Mevcut simülasyon program donanımlarının amacı, kullanımı kolay ve "Bütünleşik Simülasyon

Çevresi" sağlamaktadır. Bu cins bir bütünleşik çevrede; model tasarımı, modellerin geçerliliğinin sınanması, giriş verileri hazırlama, çıkış sonuçlarının analizi, deneyleri tasarlama gibi birçok modül entegre olarak çalışır⁸⁴.

⁸⁴ Simülasyon Dilleri Ve Kullanılan Paket Programlar, 31 Ekim 2005, <http://www.elektrotekno.com/about20844.html>, (30 Ocak 2008) , s: 1-2

4. MONTE CARLO SİMÜLASYON METODU

4.1. Monte Carlo Simülasyon Metodunun Tarihçesi

Monte Carlo simülasyonu, ilk olarak 1940'lı yıllarda Los Alamos Laboratuvarında nükleer silah geliştirmeye çalışan bilim adamlarının bu metodu bir dizi matematiksel yöntemle uygulamasıyla ortaya çıktı. Metodun aslı şans oyunlarındaki davranış tarzı ve bunun sonuçlarına bakarak denemelerin yapıldığı ilginç bir olaya dayanır. Esas olarak bilgisayarlarla bir ilgisinin olmamasına rağmen modern bilgisayarların gelişmesiyle bu nümerik ve simülasyona dayalı oyunlara ilgi artmıştır.

Bu yöntemin, aslında bir şans oyununa çözüm arama veya rassal örneklemeyle ilgili olmasına rağmen çok daha önemli etkiler doğurması da ilginçtir. Bazı yazarlara göre de Monte Carlo yöntemi kabaca nümerik değerler üreten bir yöntemden ileriye gidememiştir⁸⁵.

Monte Carlo Simülasyon tekniğinin ortaya çıkışı ilginç bir tesadüfe dayanır. Teknik, 1946 yılında Stanislaw (Stan) Ulam isimli bilim adamının 52 adet kağıtla oynanan soliter oyunundan esinlenmesi ile başlar. Oyunu başarıyla bitirmek için şansını hesaplamaya başlayan Ulam, pek çok deneme yapmış ve sonunda gerçekten oynayıp vakit kaybetmek yerine başarı şansını hesaplamak için çözüme götürecek pratik bir metod kullanma fikri üzerinde yoğunlaşmıştır. Daha sonra ise buna benzer bir tekniği nötron difüzyonunda ve diğer matematiksel fizik konularında kullanılabileceği fikrini meslektaşı John Von Neumann ile paylaşması ve 1947 Mart'ında Los Alamos Scientific Laboratory'deki çalışmaları Monte Carlo metodunun ve modern anlamda simülasyonun miladı kabul edilir⁸⁶.

İlk olarak nükleer silah geliştirilmek için kullanılan bu yöntem, adını çıktığı noktadan, yani Ulam'ın kağıt oyunuyla yaptığı denemeden alır. 2. Dünya Savaşı sürerken büyük bir gizlilik içinde yürütülen bu çalışmaya adını veren, kumarhaneleriyle ünlü Monaco'nun Monte Carlo şehridir.

⁸⁵ Kalos, Paula, s: 1

⁸⁶ Roger Eckhardt, Stan Ulam, John Von Neumann, And The Monte Carlo Method, Los Alamos Science Special Issue, 1987, <http://library.lanl.gov/cgi-bin/getfile?00326867.pdf>, s: 131

4.2. Monte Carlo Simülasyon Metodunun Tanımı

Günümüzde bilgisayarların yaygınlaşması sayesinde rassal sayılar da pek çok alanda kullanılmaya başlanmıştır. Örneğin bilgisayar oyunları ve testlerin yapılması için yapay bilgi setleri oluşturmak bunların başlıcalarıdır. Fakat Monte Carlo dendiğinde akla sadece rassal sayı üretimi gelmemelidir. Monte Carlo yöntemiyle bizi asıl ilgilendiren stokastik bir sürecin analizi için rassal sayılar üretmek ve üzerinde çeşitli hesapların ve analizlerin yapılmasını sağlamaktır. Rassal süreçle kastedilen ise oluşumu ve gelişimi rassal, diğer bir deyişle şansa bağlı gerçekleşen bir olaylar dizisidir⁸⁷.

Çoğu zaman Monte Carlo ve simülasyon yöntemi arasında şu şekilde bir ayırım yapılır; simülasyon doğal stokastik süreçlerin direkt kopyasıdır, Monte Carlo ise olasılıksal olmayan bir problemi olasılıklı bir problem olarak çözümleme yöntemidir. Bu ayırımın yapılması bazen yararlı olabilir ancak çok fazla sürdürülmesi de çoğu zaman imkansız olacaktır. Örneğin radyasyonun atomlar tarafından emilmesi ve cisimlerle olan etkileşimi, her iki olayın da tahmin edilemeyecek derecede belirsizlik içermesine rağmen doğal stokastik proses olarak tanımlanır⁸⁸.

Monte Carlo simülasyonunda bir deneyin sonucunu tahmin etmek için rasgele örneklemeden yararlanılır. Bu yöntem günümüz simülasyonunun önde gelen yöntemidir⁸⁹.

Sistemin durumunu değiştiren olayların gerçekleşme zamanlarına ait değerlerin verilen bir olasılık dağılımından belirlenmesi, olasılık dağılımından örnek seçme, rasgele değer üretimi veya Monte Carlo örnekleme olarak bilinir. Kesikli dağılımlardan örnek seçiminde olasılığın frekansa bağlı tanımı dikkate alınır ve uzun dönemde sonuçların olasılık dağılımındaki olasılıkların belirlediği sıklıklarda gerçekleşmesi beklenir⁹⁰. (Örneğin hilesiz zarın atılması deneyinde her bir yüzeyin gelmesi olasılığı uzun dönemde 1/6 , aynı şekilde hilesiz para deneyinde herhangi bir yüzeyin gelme olasılığı 1/2 olarak gerçekleşir)

⁸⁷ Kalos ve Paula, s: 2

⁸⁸ Kalos ve Paula, s: 3

⁸⁹ Taha, s: 666

⁹⁰ Cinemre, s: 647

4.3. Rassal Sayıların Üretilmesi Ve Test Edilmesi

Simülasyon deneylerinin yapılabilmesi için veri toplanması ve bu veri seti üzerinde çalışma yapılması gerekmektedir. Bu veri setini araştırmacı kendisi gözlem yaparak toplayabileceği gibi daha önce yapılan araştırmalardaki gözlemlerden de faydalanabilir. Araştırması yapılan konu gerçekte henüz varolmayan bir sistemle ilgiliyse ya da mevcut veri setindeki gözlemlerden çok daha fazla sayıda veriye ihtiyaç duyulursa o zaman rassal sayı üretme tekniğine ihtiyaç duyacaktır.

Monte Carlo simülasyonu bir olasılık dağılımından rasgele denemeler sağlamak için rasgele sayıları gerektirir. Rasgele bir sayının gerçekleşme olasılığı dizideki herhangi bir sayının gerçekleşme olasılığı ile aynıdır. Rasgele sayılar;

- 1) El işlemleriyle
- 2) Tablolarla
- 3) Bilgisayar yöntemleri ile

elde edilebilir.

El işlemleri yöntemi yorucu olması nedeniyle pratik değildir. Rulet kağıtlarının karılması, çanaktaki boncuk veya toplar, para atışı, zar atışları kullanılan araçlar olacaktır⁹¹.

Monte Carlo simülasyonu statik simülasyon modelidir. Sistemin belirli zaman aralığında belirli bir anındaki durumunu yansıtmak amacıyla kullanılırlar.

Monte Carlo örnekleme ile kesikli rasgele değişken için rasgele değer üretimi;

- 1) Rasgele değişken için birikimli olasılık dağılımının oluşturulması
- 2) Rasgele sayıların değişkenin değişik değerlerine paylaştırılmasında birikimli olasılık dağılımının kullanılması

⁹¹ Halaç, s: 645

olmak üzere iki adımdan oluşur⁹².

RAND tarafından üretilen rassal sayı tabloları Monte Carlo deneylerini içeren ekonometrik ve mühendislik alanlarında yapılan denemeler için artık çok sık kullanılan standart tablolar haline gelmiştir. Başta Monte Carlo deneyi olan çok sayıdaki olasılıksal deney içeren problemin çözümü için The RAND Corporation'a ihtiyaç duyulması ve talebin artması sebebiyle rasgele sayıların üretilmesine ihtiyaç duyulmuştur. Bu ihtiyacı karşılamak için de yüksek kalitede rasgele sayı tabloları üretilmiştir⁹³.

Sözde rassal sayıları üretmek için güvenilir bir yöntem oluşturulmadan önce tamamen fiziksel süreçlerle üretilen ve rassal sayıları içeren tablolar bulunmaktaydı. Bu tabloların en geniş biçimde kullanılanı tamamen fiziksel süreçlerle oluşturulan ve 1.000.000 rassal sayı içeren 'RAND Tabloları' (The RAND Corporation, 1955)'ydi. Marsaglia (1995), 60 ayrı dosyada 4.800.000 karakter içeren bir CD-ROM üretti. Bu karakterler iki ya da daha fazla sayıdaki deterministik sayı üreticinin birleştirilmesiyle ve üç adet fiziksel süreçleri kullanan aygıt tarafından üretildi⁹⁴.

Bu rasgele sayılar fiziksel süreçler ile (Elektrik akımı gibi) üretilirler ve gerçekten rasgele sayılar olduğu düşünülürler.

Rasgele sayıları elde etmek için genel süreç sözde rasgele sayıları üretmektir. Bu işlem genellikle bilgisayar programı ile olur. Sözde rasgele sayı dizisi tamamen matematiksel süreçle elde edildiğinden gerçekte tam rasgele değildirler. Buna bağlı olarak bu yolla üretilen sayılar rasgele olmaları için belli istatistiksel testlerden geçtikten sonra rasgele olarak işlemlerde kullanılmaktadırlar⁹⁵.

Dijital bilgisayarlar tam olarak rassal sayı üretemezler, bu yüzden bilgisayarları rassal olayların dışsal kaynaklarıyla ilişkilendirmek uygun olmayabilir. İstatistik, mühendislik ve doğa bilimlerindeki pek çok uygulamada, bu sözde rasgele sayıların

⁹² Cinemre, s: 653

⁹³ The RAND Corporation, **A Million Random Digits With 100000 Normal Deviates**, The RAND Corporation, 1955, http://rand.org/pubs/monograph_reports/MR1418/index2.html, 30 Ocak 2008, s:1

⁹⁴ James E. Gentle, **Random Number Generation And Monte Carlo Methods**, 2. Basım, New York - ABD, Springer, 2003, s:46

⁹⁵ Halaç, s: 466

kaynağı biliniyorsa ve bu kaynak rassal olarak dağılmış bir dağılımdan elde edilmişse bu bir dezavantaj yaratmaz⁹⁶.

Monte Carlo Simülasyon denemelerinin esası rassal sayı üreteçleridir, rassal sayı üreteçlerinin esası ise düzgün dağılmış rassal sayı üreteçleridir. Rassallığın dayandığı konsept ilginçtir çünkü doğada hiçbir olay tam ve kesin olarak rassal değildir. Bu olayları etkileyen faktörleri tam olarak bilemeyiz; bu yüzden sınırlı bilgimiz dahilinde bu olayların rassalmış gibi görüldüğünden bahsederiz. Benzer şekilde bilgisayarlarda üretilen rassal sayılar da aslında gerçek olmayan – sözde rassal - rassallığa yakın sayılardır. Çünkü belli deterministik algoritmalarla üretildiklerinden sadece rassal gibi görünürler. Buna rağmen bu sayıları problemin ihtiyaç duyduğu rassallık özelliklerini sağlıyorsa kullanabiliriz⁹⁷.

4.4. Rasgele Sayı Üretme Yöntemleri

4.4.1. Uygunluk Yöntemi

D. H. Lehmer, 1948 yılında rassal sayı üretmek için Basit Lineer Uygunluk Üretecini (Simple Linear Congruential Generator) önermiştir. Bu üreteçte her sayı kendisinden sonra gelecek olan sayıyı basit bir lineer fonksiyonun modunun alınmasıyla tayin eder. Bu üreteç her ne kadar uzun sayı akımları üretemese de uniform dağılıma benzeyen bağımsız rasgele sayılar üretebilmeyi gerçekleştirmiştir. Bu yöntemin özelliklerinin kullanılmasıyla daha iyi ve daha uzun sayı akımları üretilebilir⁹⁸.

Fonksiyonun formu şu şekildedir;

$$\chi_i \equiv (a\chi_{i-1} + c) \bmod m \quad , \quad 0 \leq \chi_i < m$$

Bu denkliğin değişkenlerini,

⁹⁶ Gentle, s: 2

⁹⁷ Xitao Fan, Akos Felsovalyi, Stephan A. Sivo, Sean C. Kenan, **SAS For Monte Carlo Studies A Guide For Quantitative Researchers**, 1. Basım, Carolina – ABD, SAS Institute Inc., 2002, s: 26

⁹⁸ Gentle, s: 11

a = Çarpan
 c = Eklenti
 m = Mod

olarak tanımlayabiliriz.

Denklikte c sayısı 0' a eşit olarak da alınabilir. Bu durumda bu denklik Çarpan Uygunluk Yöntemi adını alır.

$$\chi_i = a\chi_{i-1} \bmod m \quad , \quad 0 < \chi_i < m \quad 2.1$$

c 'nin 0 dan farklı olduğu durumlardaysa yönteme Karışık Uygunluk Yöntemi adı verilir. Bu üreteçte bir tek tohum değeri vardır ve o da X_0 'dır. Bu yönteme Lehmer Dizisi adı verilir. Her bir X_i sayısı m sayısına bölünerek (0,1) aralığında sayılar elde edilir.

$$u_i = \chi_i / m$$

a ve m sayılarının uygun seçilmesiyle u_i sayıları (0,1) aralığında rassal ve düzgün dağılmış sayılar elde edilir. 2.1'deki denkliğin tamsayılar için aşağıdaki denklige eşit olduğunu söyleyebiliriz.

$$u_i \equiv au_{i-1} \bmod 1 \quad , \quad 0 < u_i < 1$$

Bu yineleme, otoregresif modeller için üretilen ilk denkleme benzemektedir.

$$U_i = pu_{i-1} + E_i$$

p , otoregresif modelin katsayısı, E_i , (0,1) aralığında rassal dağılmış rassal değişkendir⁹⁹.

Bir rasgele sayı üreticinin iyi olabilmesi için aşağıdaki özellikleri taşıması gerekmektedir.

⁹⁹ Gentle, s: 11

- 1) Üretilen sayılar (0,1) aralığında olabildiğince tekdüze dağılmalıdır.
- 2) Üretilen sayılar birbirinden bağımsız olmalı, üretilen sayının değeri kendisinden önce ve sonra üretilen sayılardan etkilenmemelidir.
- 3) Üretilen sayılar, istenildiği takdirde tekrar üretilebilir olmalıdır.
- 4) Üretilen bir rasgele sayı uzun bir dönem bir daha ortaya çıkmamalıdır. Üretilen rasgele sayılar dizisinde aynı rasgele sayıya iki kez raslanmamalı, kısaca sayılar birbirinden farklı olmalıdır.
- 5) Sayılar hızlı üretilmeli ve sayı üretmede kullanılan program bilgisayar belleğinde fazla yer kaplamamalıdır.
- 6) Üreteç 0 üretmemelidir¹⁰⁰.

4.4.2. Ters Dönüşüm Yöntemi

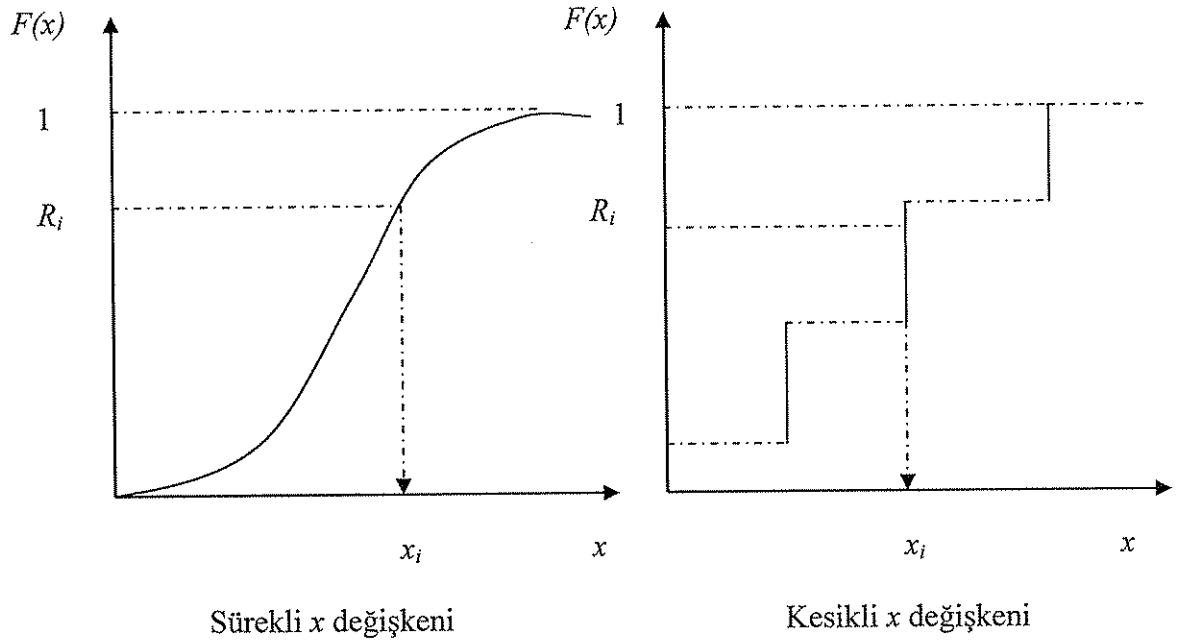
$f(x)$ olasılık yoğunluk fonksiyonundan (Sürekli ya da kesikli) bir x rasgele örneklemi elde edileceğini varsayalım. Ters Dönüşüm Yöntemi, önce y 'nin tanımlanmış tüm değerleri için $0 \leq F(x) \leq 1$ olmak üzere $F(x) = P\{y \leq x\}$ kümülatif yoğunluk fonksiyonunun kapalı bir formunu belirlemektedir. R , uniform bir (0,1) dağılımından elde edilen rasgele bir değişken olarak verilmişse ve F^{-1} de F 'nin tersi ise, yöntemin adımları aşağıdaki gibi olacaktır¹⁰¹:

Adım 1: $R(0,1)$ rasgele sayısını üret

Adım 2: İstenen $x = F^{-1}(R)$ örneklemi hesapla.

¹⁰⁰ Cinemre, s: 653

¹⁰¹ Taha, s: 676



Şekil 8. Ters Dönüşüm Yöntemi

Önerilen bu yöntemin geçerliliğinin aşağıdaki teoremdе gösterildiği gibi $0 \leq z \leq 1$ aralığında uniform dağılmış $z=F(x)$ rasgele değişkenine dayandığı görölmektedir.

Teorem: $-\infty \leq x \leq \infty$ olmak üzere, x rasgele değişkeninin $F(x)$ kümülatif yoğunluk fonksiyonu verildiğinde, $0 \leq z \leq 1$ olmak üzere $z=F(x)$ rasgele değişkeni aşağıdaki olasılık yoğunluk fonksiyonuna sahip olur:

$$f(z)=1 \quad \rightarrow \quad 0 \leq z \leq 1$$

Bu bir uniform (0,1) dağılımıdır.

İspat: Rasgele değişken sadece ve sadece

$$P\{z \leq Z\} = Z \quad , \quad 0 \leq Z \leq 1$$

ise uniform dağılmıştır. Bu sonuç, aşağıdaki eşitliklerden de doğrudan çıkarılabilir:

$$P\{z \leq Z\} = P\{F(x) \leq Z\} = P\{x \leq F^{-1}(Z)\} = F[F^{-1}(Z)] = Z$$

Ayrıca, $0 \leq P\{z \leq Z\} \leq 1$ olduğundan $0 \leq Z \leq 1$ 'dir¹⁰².

4.4.2.1. Ters Dönüşüm Yöntemini Kullanarak Bilinen Olasılık Dağılımlarından Rassal Değişkenlerin Üretilmesi

Ters dönüşüm yöntemiyle bilinen olasılık dağılımlarından rassal değişkenler türetmek yoluyla rassal sayılar elde edilebilir. Uygunluk yöntemlerinden farklı olarak bu yöntemlerle sadece rassal sayılar değil rassal sayıları elde etmek için yeni rassal değişkenler de türetilir.

4.4.2.1.1. Üssel Dağılım

Aşağıdaki üssel dağılıma ait olasılık yoğunluk fonksiyonunu rassal sayıları üretmek için ters dönüşüm yöntemine göre kullanalım;

$$f(x) = \lambda e^{-\lambda x} \quad , \quad 0 \leq x < \infty$$

birikimli olasılık fonksiyonu;

$$F(x) = \int_0^x \lambda e^{-\lambda u} du = 1 - e^{-\lambda x} \quad , \quad x \geq 0 \text{ ve buradan,}$$

$$X = F^{-1}(R) = -\frac{1}{\lambda} \ln(1 - R)$$

Bu denklem ters dönüşüm yöntemini üssel dağılmış bir değişkene nasıl uygulandığını gösterir. R ve $(1-R)$ aynı olasılık fonksiyonuna sahip olduklarından $(1-R)$ yerine R kullanırsak,

$$X = -\frac{1}{\lambda} \ln(R)$$

elde edilmiş olur.

¹⁰² Taha, s: 676

Örneğin $\lambda = 1/3$ parametresiyle üssel dağılmış bir değişkenden rassal sayı elde etmek için örnekleme yaparsak;

$X = -3 \ln R$, eğer bir sonraki rassal sayı, $R=0,1367$ ise üssel dağılmış değişkenin değeri,

$$X = -3 \ln(0,1367) = 5,970 \text{ olarak elde edilir}^{103}.$$

4.4.2.1.2. Earlang Dağılımı

Üssel Dağılımın bu özelliği kullanılarak Earlang Dağılımından rassal değişken türetilir. Ortalaması $1/k\alpha$ olan k sayıdaki üssel değişkenin toplamı da k parametrelili Earlang dağılımına uyar. Bu yüzden 0 ve 1 aralığında verilen k sayıdaki rassal değişkeni türetmek için aşağıdaki formül kullanılır;

$$x = \sum_{i=1}^k \frac{\ln(1-r_i)}{-k\alpha} \quad , \quad i = 1, 2, \dots, k$$

Aşağıdaki kısaltma yapılırsa;

$$x = -\frac{1}{k\alpha} \ln \left(\prod_{i=1}^k (1-r_i) \right) \quad , \quad i = 1, 2, \dots, k$$

olarak elde edilir. Burada da üssel dağılımda olduğu gibi $(1-r_i)$ yerine r_i de yazılabilir¹⁰⁴.

4.4.2.1.3. Uniform Dağılım

Verilen bir $[a,b]$ aralığında düzgün dağılan bir x değişkenine ait olasılık yoğunluk fonksiyonu;

$$f(x) = \begin{cases} \frac{1}{b-a} & , \quad a \leq x \leq b \\ 0 & , \quad \text{Diğer durumlarda} \end{cases}$$

¹⁰³ J. S. Dagpunar, **Simulation And Monte Carlo With Applications In Finance And MCMC**, Chichester – İngiltere, John Wiley&Sons Ltd., 2007, s: 37 - 38

¹⁰⁴ Hillier, Lieberman, s: 870

şeklindedir. Aşağıdaki adımlar izlenerek uniform dağılmış bir değişkenden rassal sayılar elde edilir.

Adım 1: Aşağıdaki şekilde birikimli olasılık yoğunluk fonksiyonunu üret.

$$F(X) = \begin{cases} 0 & , \quad x < a \\ \frac{x-a}{b-a} & , \quad a \leq x \leq b \\ 1 & , \quad x > b \end{cases}$$

Adım 2: $F(X) = \frac{X-a}{b-a} = R$ olacak şekilde düzenle.

Adım 3: Bu denklem çözülerek $X = a + (b-a)R$ şeklinde elde edilir¹⁰⁵.

4.4.2.1.4. Weibull Dağılımı

Weibull Dağılımı, bir makinede veya makinenin bileşenlerinde meydana gelebilecek hasarlar için geçen zamana ilişkin bir olasılık dağılımıdır. Weibull dağılımında konum parametresi olan ν parametresi 0' eşit kabul edildiğinde Weibull dağılımı için olasılık yoğunluk fonksiyonu;

$$f(x) = \begin{cases} \frac{\beta}{\alpha^\beta} x^{\beta-1} e^{-(x/\alpha)^\beta} & , \quad x \geq 0 \\ 0 & , \quad \text{Diğer durumlarda} \end{cases}$$

şeklindedir. Bu dağılımda verilen $\alpha > 0$ ve $\beta > 0$ parametreleri sırasıyla ölçek ve şekil parametreleridir. Aşağıdaki adımlar izlenerek Weibull dağılan değişken için rassal değişken üretilir;

Adım 1: Birikimli olasılık yoğunluk fonksiyonunu hesaplanır.

$$F(X) = 1 - e^{-(x/\alpha)^\beta} , \quad x \geq 0$$

¹⁰⁵ Jerry Banks, John S. Carson II, Barry Nelson ve David M. Nicol, **Discrete Event System Simulation**, 3. Basım, New Jersey – ABD, Prentice Hall International Inc., 2001, s: 294

Adım 2: $F(X) = 1 - e^{-(x/\alpha)^\beta} = R$ çözümüne ulaşılmaya çalışılır.

Adım 3: Buradan

$$X = \alpha[-\ln(1-R)]^{1/\beta}$$

olarak elde edilir¹⁰⁶.

4.4.2.1.5. Cauchy Dağılımı

Normal dağılıma benzer fakat merkezde daha ince ve kuyruk kısımlarında daha geniş bir alana yayılmıştır. Olasılık yoğunluk fonksiyonu

$$f(x) = \frac{b}{\pi(b^2 + (x-a)^2)} \quad , \quad -\infty < x < \infty$$

ve birikimli olasılık yoğunluk fonksiyonu,

$$F(X) = \frac{1}{2} + \frac{1}{\pi} \arctan\left(\frac{x-a}{b}\right) \text{ şeklindedir. Buradan } X \text{ değişkeni;}$$

$$X = a + b \tan\left\{\pi\left(R - \frac{1}{2}\right)\right\} \text{ veya}$$

$$X = a + \frac{b}{\tan(\pi R)}$$

olarak elde edilir¹⁰⁷.

4.4.2.1.6. Geometrik Dağılım

Verilen bir geometrik dağılıma ait bir olasılık yoğunluk fonksiyonu,

$$f(x) = p(1-p)^x \quad , \quad x = 1, 2, 3, \dots$$

ve birikimli olasılık yoğunluk fonksiyonu;

¹⁰⁶ Banks, Carson II, Nelson ve Nicol, s: 294

¹⁰⁷ Don L. McLeish, **Monte Carlo Methods In Finance**, İngiltere, John Wiley&Sons Ltd., 2000, s: 57

$$F(X) = 1 - (1 - p)^x, \quad x \geq 0$$

şeklindedir. Buradan ters dönüşüm yöntemiyle rassal değişken,

$$X = 1 + \left(\frac{\log(1 - R)}{\log(1 - p)} \right) \text{ olarak elde edilir}^{108}.$$

4.4.2.1.7. Pareto Dağılımı

Pareto dağılımı, ekonometrik araştırmalarda alt sınırı 0'a çok yakın miktarları (b) gösteren modellemelerde kullanılır.

Birikimli olasılık yoğunluk fonksiyonu

$$F(X) = 1 - \left(\frac{b}{x} \right)^a, \quad x \geq b \geq 0$$

olan bir pareto dağılımından rassal dönüşüm yöntemi kullanılarak aşağıdaki rassal değişken elde edilir¹⁰⁹;

$$X = \frac{b}{(1 - R)^{1/a}}$$

veya

$$X = \frac{b}{R^{1/a}}$$

4.4.2.1.8. Normal Dağılım

Merkezi Limit Teoremi kullanılarak normal dağılımdan rassal değişken üretilebilir. Çünkü elde edilen rassal sayılar 0 ve 1 aralığında, 1/2 ortalama ve $1/\sqrt{12}$ varyansla düzgün dağılıma uyar. Bu teorem, n sayıdaki düzgün dağılmış rassal

¹⁰⁸ McLeish, s: 58

¹⁰⁹ McLeish, s: 58

değişkenin toplamının $n/2$ ortalama ve $\sqrt{n/12}$ varyansla yaklaşık olarak normal dağıldığını varsayar. $r_1, r_2, \dots, r_n$ düzgün dağılmış gözlemler olmak üzere,

$$x = \frac{\sigma}{\sqrt{n/12}} \sum_{i=1}^n r_i + \left(\mu - \frac{n\sigma}{2\sqrt{n/12}} \right), \quad i = 1, 2, \dots, n$$

μ ortalamalı ve σ standart sapmalı yaklaşık normal dağılmış değişkenler elde edilir. Bu tahmin yöntemiyle n 'in küçük değerleri için bile yeterince iyi gözlem değerleri elde edilir¹¹⁰.

Bu yöntemin dezavantajı her bir normal örneklem için 12 tane (0,1) rassal sayı üretmek gerekmektedir. Bu da verimsiz bir durum olmaktadır. Çok daha verimli bir yöntem,

$$x = \sqrt{-2 \ln(R_1)} \cos(2\pi R_2)$$

dönüşümünü gerektirmektedir. Box ve Muller, x 'in Standard bir $N(0,1)$ olduğunu kanıtlamışlardır.

Box ve Muller, yukarıdaki formülde $\cos(2\pi R_2)$ yerine $\sin(2\pi R_2)$ yazıldığında başka bir $N(0,1)$ düzgün değişkeni elde edileceğini göstermişlerdir. Bu da R_1 ve R_2 gibi iki rasgele sayı kullanarak aynı anda iki tane $N(0,1)$ düzgün dağılmış örneklemının üretilmesi demektir¹¹¹.

4.4.2.1.9. X^2 Dağılımı

Ki Kare dağılımını incelemek için, standardize edilmiş normal dağılımların toplamı olduğu gerçeğini kullanabiliriz. $y_1, y_2, \dots, y_n$ 0 ortalama ve 1 varyansla normal dağılmış değişkenler olmak üzere

$$x = \sum_{i=1}^n y_i^2, \quad i = 1, 2, \dots, n$$

¹¹⁰ Hillier, Lieberman, s: 871

¹¹¹ Taha, s: 680

değişkeni de n serbestlik dereceli Ki Kare dağılımından elde edilen rassal değişkendir¹¹².

4.4.3. Kabul – Red Yöntemi

Kabul – Red Yöntemi, diğer yöntemlerin uygulanamadığı karmaşık olasılık yoğunluk fonksiyonlarını kullanmak üzere tasarlanmıştır. Yöntemin genel düşüncesi, karmaşık olasılık yoğunluk fonksiyonu $f(x)$ 'i daha analitik olarak kullanabilen bir temsili olasılık yoğunluk fonksiyonu $h(x)$ ile değiştirmektir. $h(x)$ 'ten alınan örneklemeler daha sonra orijinal olasılık yoğunluk fonksiyonu $f(x)$ 'ten örneklemeler belirlemede kullanılmaktadır¹¹³.

$g(x)$ büyütme fonksiyonu, yetişebildiği her yerde $f(x)$ üzerinde baskın bir davranış sergilemektedir. Şöyle ki;

$$g(x) \geq f(x) \quad , \quad -\infty < x < \infty$$

olup bunun ardından temsili olasılık yoğunluk fonksiyonu olan $h(x)$, $g(x)$ 'in normalizasyonu ile

$$h(x) = \frac{g(x)}{\int_{-\infty}^{+\infty} g(x) dx} \quad , \quad -\infty < x < \infty$$

şeklinde tanımlanır. Kabul – Red Yönteminin adımları aşağıdaki gibidir:

Adım 1: Ters dönüşüm yöntemini kullanarak $h(x)$ 'ten $x=x_1$ örneklemini elde et.

Adım 2: Bir (0,1) rasgele sayısı elde et.

Adım 3: Eğer, $R \leq \frac{f(x_1)}{g(x_1)}$ ise, x_1 'i $f(x)$ 'in uygun bir örneği olarak kabul et.

Aksi halde, x_1 'i reddederek adım 1'e dön.

¹¹² Hillier, Lieberman, s: 1109

¹¹³ Taha, s: 682

Yöntemin geçerliliği aşağıdaki eşitliğe dayanmaktadır:

$$P\{x < a, x = x_1 \text{ kabul edilmiştir}, -\infty < x_1 < \infty\} \rightarrow \int_{-\infty}^a f(x)dx, \quad -\infty < a < \infty$$

Bu olasılık ifadesine göre, adım 3'teki koşulu sağlayan $x=x_1$ örnekleme, istendiği gibi orijinal olasılık yoğunluk fonksiyonu $f(x)$ 'ten alınan bir örneklem olmaktadır.

Önerilen bu yöntemin etkinliği, adım 3'teki red olasılığının azalmasıyla birlikte artmaktadır. Bu olasılık, büyütme fonksiyonu $g(x)$ 'in özel seçimine bağlı olup, $f(x)$ ile çok uyumlu bir $g(x)$ 'in seçimiyle azalacaktır¹¹⁴.

4.4.3.1. Kabul – Red Yöntemi Yardımıyla Bilinen Olasılık Dağılımlarından Rassal Değişken Üretilmesi

4.4.3.1.1. Poisson Dağılımı

x Poisson rassal değişkeni $\lambda > 0$ ortalamayla aşağıdaki olasılık fonksiyonuna sahiptir;

$$f(x) = \frac{e^{-\lambda} \lambda^x}{x!}, \quad x = 0, 1, 2, \dots$$

x , burada birim zamanda sisteme Poisson süreciyle gerçekleşen gelişlerin sayısını göstermektedir. $A_1, A_2, A_3, \dots$ şeklinde gösterilen ardışık gelişler arası süre λ ortalamayla Poisson dağılımına uymaktadır. Buna ek olarak daha önce üssel dağılım Poisson dağılımından türetilir. Bu yüzden üssel dağılım ve Poisson dağılımları arasındaki bu ilişki kullanılarak aşağıdaki eşitsizlikler kullanılabilir.

$N=n$ eşitliği aşağıdaki koşullar sağlandığı durumda doğru kabul edilir.

$$A_1 + A_2 + \dots + A_n \leq 1 < A_1 + A_2 + \dots + A_n + A_{n+1}$$

¹¹⁴ Taha, s: 682

n . varış 1. zamandan önce, $(n+1)$. Varış 1. zamandan sonra gerçekleşmektedir. Üssel dağılım için daha önce kullandığımız eşitliği burada da kullanırsak,

$$(A_i = \left(\frac{1}{\lambda}\right) \ln R_i)$$

$$\sum_{i=1}^n -\frac{1}{\lambda} \ln R_i \leq 1 < \sum_{i=1}^{n+1} -\frac{1}{\lambda} \ln R_i$$

Eşitliğin her iki tarafını da $-\lambda$ ile çarparsak,

$$\ln \prod_{i=1}^n R_i = \sum_{i=1}^n \ln R_i \geq -\lambda > \sum_{i=1}^{n+1} \ln R_i = \ln \prod_{i=1}^{n+1} R_i$$

$e^{\ln x} = x$ ilişkisi kullanılarak

$$\prod_{i=1}^n R_i \geq e^{-\lambda} > \prod_{i=1}^{n+1} R_i$$

Aşağıdaki adımlar izlenerek kabul – red yöntemiyle rassal değişken elde edilir.

Adım 1: $n=0$ ve $P=1$ olarak belirle.

Adım 2: R_{n+1} sayısını üret ve P yerine $P.R_{n+1}$ yazılır.

Adım 3: $P < e^{-\lambda}$ ise $N=n$ eşitliğini kabul et. Aksi halde geçerli olan n 'i reddet, n 'i 1'e yükselt ve 2. adıma geri dön¹¹⁵.

4.4.3.1.2. Gamma Dağılımı

Gamma dağılımı yardımıyla rassal değişken oluşturmak için pek çok kabul – red tekniği geliştirilmiştir. En etkili olanı, verilen şekil parametresi her bir $\alpha \geq 1$ değeri için ortalama deneme sayısının 1.13 ve 1.47 arasında olan Cheng (1977) tarafından geliştirilen tekniktir.

¹¹⁵ Banks, Carson II, Nelson, Nicol, s: 314

$$f(x) = \frac{(\lambda x)^{\alpha-1} \lambda e^{-\lambda x}}{\Gamma(\alpha)}, \quad x > 0$$

şeklinde verilen bir gamma dağılımı için diğer bir alternatif de şekil parametresinin $\alpha = k$ gibi bir tam sayı olması durumunda gamma dağılımının özel bir hali olan Earlang dağılımı için konvolüsyon tekniğini kullanmaktır. Bu yöntem genel olarak şekil parametresi α ve ölçek parametresi λ olan, $1/\lambda$ ortalamalı ve $1/\alpha^2$ varyanslı değişkenler üretir.

Adım 1: $a = (2\alpha - 1)^{1/2}$, $b = 2\alpha - \ln 4 + 1/a$ değerleri hesaplanır.

Adım 2: R_1 ve R_2 sayıları üretilir.

Adım 3: $X = \alpha \left[\frac{R_1}{(1 - R_1)} \right]^a$ hesaplanır.

Adım 4a: $X > b - \ln(R_1^2 R_2)$ ise X 'i reddet ve 2. adıma dön.

Adım 4b: $X \leq b - \ln(R_1^2 R_2)$ ise X 'i arzu edilen değişken olarak kullan. Bu adımda üretilen değişkenin varyansı ve ortalaması α 'dır.

Adım 5: X yerine $X/\alpha\lambda$ yaz.

3. adımda üretilen değişken gamma dağılmaz; ancak X 'in 4a adımında reddedilen değerleri 4b adımında kabul edilen değerlerinin gamma dağıldığını garanti eder¹¹⁶.

¹¹⁶ Banks, Carson II, Nelson, Nicol, s: 315

4.4.3.1.3. Üçgen Dağılım

Üçgen dağılıma ait olasılık yoğunluk fonksiyonu aşağıdaki şekilde verilmiş olsun;

$$f(x) = \begin{cases} x & , \quad 0 \leq x \leq 1 \\ 1-(x-1) & , \quad 1 \leq x \leq 2 \\ 0 & , \quad \text{diğer durumlarda} \end{cases}$$

Kabul – red yöntemiyle rassal değişken üretmek için aşağıdaki iki adım izlenir;

Adım 1 : 0 ve 1 aralığında r_1 rassal sayısını üret, $x=2r_1$ olarak düzenle

Adım 2 : Aşağıdaki durumda x' , kabul et.

$$p = \begin{cases} x & , \quad 0 \leq x \leq 1 \\ 1-(x-1) & , \quad 1 \leq x \leq 2 \end{cases}$$

Diğer durumlarda x' i reddet ve bu iki adımı tekrar et. x' in kabul veya reddedilmesi görüldüğü gibi bu olasılığa bağlıdır ve süreç şu şekilde devam eder:

Adım 3 : 0 ve 1 aralığında r_2 rassal sayısını üret.

$r_2 \leq f(x)$ ise x' i kabul et.

$r_2 > f(x)$ ise x' i reddet.

x eğer reddedilmişse iki adımı tekrar et.

$x = 2r_1$ rassal değişkeni, $f(x)$ olasılık yoğunluk fonksiyonuna göre kabul edildiği için, kabul edilen değişkenlerin olasılık yoğunluk fonksiyonu da $f(x)$ 'in özelliklerine sahiptir. Böylece kabul edilen değişkenler de $f(x)$ fonksiyonundan türetilmiş ve geçerli rassal gözlemler olarak kullanılır¹¹⁷.

¹¹⁷ Hillier, Lieberman, s: 1110

4.4.4. Konvolüsyon Yöntemi

Konvolüsyon Yönteminin temel düşüncesi, istenen örneğin kolay örneklenen diğer rassal değişkenlerin toplamı olarak ifade edilmesidir. Bu dağılımların en tipik örnekleri Earlang ve Poisson dağılımları olup üssel dağılım örneklerinden elde edilirler¹¹⁸.

4.4.4.1. Konvolüsyon Yöntemi Yardımıyla Earlang Dağılımından Rassal Değişkenlerin Türetilmesi

m tane Earlang rassal değişkeni, m tane bağımsız ve özdeş dağılmış üssel rassal değişkenlerin toplamı (Konvolüsyonu) olarak ifade edilir. Y , m Earlang dağılımını gösterebilir. Bu durumda;

$$Y = Y_1 + Y_2 + \dots + Y_m$$

olup, buradaki y_i ($i=1,2,\dots,m$)'ler bağımsız ve özdeş dağılmış üssel rasgele değişkenlerdir. Bu değişkenlerin olasılık yoğunluk fonksiyonu şöyle tanımlanır:

$$f(x) = \lambda e^{-\lambda y_i} \quad , \quad y_i \geq 0, i = 1, 2, \dots, m$$

Ters dönüşüm yöntemi kullanılarak Üssel dağılımdan rassal değişkenler,

$$y_i = -\left(\frac{1}{\lambda}\right) \ln(R_i) \quad , \quad i = 1, 2, \dots, m$$

olarak örneklenir. Dolayısıyla m tane Earlang örneği;

$$\begin{aligned} y_i &= -\left(\frac{1}{\lambda}\right) (\ln(R_1) + \ln(R_2) + \dots + \ln(R_m)) \\ &= -\left(\frac{1}{\lambda}\right) \ln(R_1 R_2 \dots R_m) \end{aligned}$$

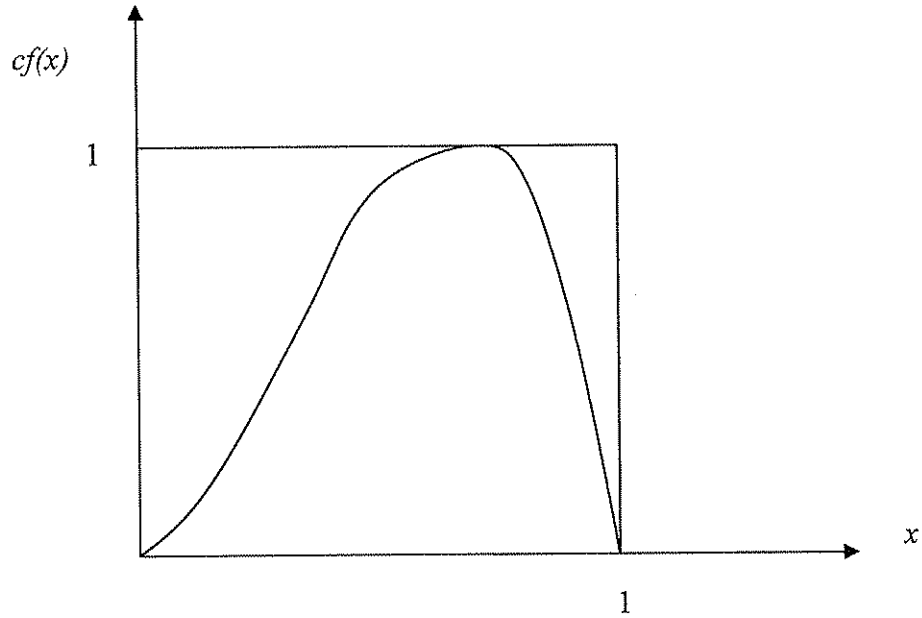
¹¹⁸ Taha, s: 678

olarak hesaplanır¹¹⁹.

4.4.5. Ayırma Yöntemi

X değişkeni, sınırlı bir (a,b) aralığında değişiyorsa ve $f(x)$ yoğunluk fonksiyonu bu aralıkta sınırlı ise aşağıdaki metot kullanılabilir.

Basit bir dönüşüm ile $f(x)$ fonksiyonu bir $c>0$ çarpanı ile çarpılarak $cf(x)$ fonksiyonu ve söz konusu aralık da birim kare içerisine alınır.



Şekil 9. Ayırma Yöntemi

Bir (u_i, v_i) rasgele sayı çifti bu kare içerisinde bir nokta belirler:

$$0 \leq u_i \leq 1, \quad 0 \leq v_i \leq 1$$

u_i değerine karşılık gelen x

$$x=a+(b-a)u_i$$

olur. Eğer,

¹¹⁹ Taha, s: 679

$$v_i \leq cf(a + (b - a)u_i)$$

ise nokta eğrinin altındadır. Bu halde u_i kabul edilir ve

$$x_i = a + (b - a)u_i$$

istenilen dağılıma uygun bir rasgele sayı olarak kabul edilir. Aksi halde u_i , v_i çifti reddedilir, yeni bir rasgele sayı alınarak işlemlere devam edilir¹²⁰.

4.5. Örnek Hacminin (Deneme Sayısının) Belirlenmesi

Monte Carlo simülasyonu deneyinde çok önemli olan bir konu da deney sayısının belirlenmesidir. Model üzerinde elle çalışılırsa deney sayısı, sarfedilecek zamanla sınırlanır. Model, bir elektronik hesap makinesine uygulanmışsa deney sayısı yüksek tutulabilir; fakat makine zamanı da pahalıdır. Bu sebeple deney sayısını güvenilir neticeler elde edilecek seviyede bırakmak zorunluluğu ortaya çıkar.

Bazı hallerde sistem hakkındaki bilgilere dayanılarak deney sayısı önceden tespit edilebilir. Diğer bazı hallerde ise elde edilen sonuçlara bakılarak deney sayısı hakkında bir hükme varılabilir. Bir grup deneyde elde edilen sonuçlar birbirinden çok farklı ise deneylere devam edilir, deney sayısı birkaç misline çıkarılır. Tersine, sonuçlar belirli bir değer etrafında toplanmışlarsa az sayıda deney güvenilir bir sonuç verebilir. Böylece elde edilen sonuç kümesinin histogramı çalışmayı yapana bu konuda rehberlik edebilir¹²¹.

Herhangi bir istatistiksel deneyde olduğu gibi n 'in büyük değerleriyle çok daha güvenilir simülasyon sonuçları elde edilebilir. Yalnız n 'in artması simülasyon deneyinin daha yüksek maliyetlerde gerçekleşmesi sonucunu da bir dezavantaj olarak beraberinde getirir¹²².

¹²⁰ Yücel, s: 37

¹²¹ Yücel, s: 17

¹²² Taha, s: 669

4.5.1. Ortalama İle Deneme Sayısının Bulunması:

Yeterince büyük örnek hacmi seçildiğini ve örneklem hacminin normal dağıldığını varsayalım. Notasyon için n örnek hacmi, σ standart sapma, $z_{\alpha/2}$ ortalaması $\alpha/2$ ile $1-\alpha/2$ arasında olan ve güvenlik seviyesi $1-\alpha$ olan normal sapma, d gerçek ortalama ve örnek ortalaması arasındaki fark olmak üzere örnek hacmi;

$$n = \frac{\sigma^2 (z_{\alpha/2})^2}{d^2}$$

formülü ile bulunur. Verilen formülün kullanılabilmesi için standart sapma, güvenlik sınırı, örnek ortalaması ile gerçek ortalama arasındaki farkın bilinmesi gerekmektedir. Halbuki standart sapma hesaplanabilir.

$$S = \sqrt{\frac{\sum_{i=1}^n (X_i - \bar{X})^2}{n-1}}$$

Ayrıca, istatistikten $z_{\alpha/2}$ yerine $t_{\alpha/2;n-1}$ konulabilir. Burada t , t dağılımıdır ve $(n-1)$ terimi de serbestlik derecesini verir. Ancak büyük örnek hacimlerinde $t_{\alpha/2;n-1}$ 'in yerine konulabileceğini yine de belirtelim, bu halde $t_{\alpha/2;n-1}$ ve n bilinmelidir ve

$$n = \frac{S^2 (z_{\alpha/2})^2}{d^2}$$

formülü kullanılabilir¹²³.

4.5.2. Standart Sapma İle Deneme Sayısının Bulunması:

Yaklaşık örnek hacmi, anakütlenin varyans duyarlılığı ve tanımlanan güvenlik seviyesi kavramlarından bulunabilir. Genel yaklaşım, $(n-1)$ serbestlik derecesi ile normalliği varsayılarak X^2 dağılımı olan

¹²³ Halaç, s: 467 - 468

$$\chi^2 = \frac{f(n-1)s^2}{\sigma^2} \quad (s^2 \text{ örnek varyansı, } \sigma^2 \text{ anakütle varyansını}$$

göstermektedir.)

bağıntısının istatistik testine göre örnek hacminin belirlenmesidir.

Bununla beraber büyük örnek hacimleri için normal dağılım olduğu gerekçesiyle örnek hacmi için

$$n = \frac{d'(n-1)}{\sqrt{2(n-1)}}$$

bağıntısı yazılır ve

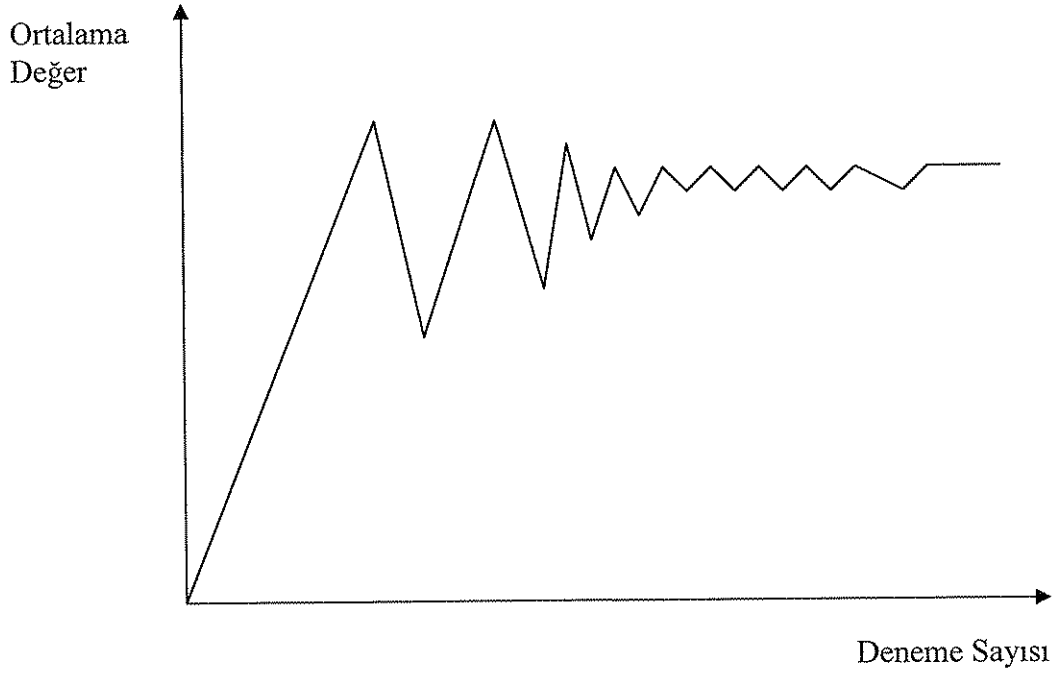
$$n = \frac{2(z_{\alpha/2})^2}{(d')^2} + 1$$

haline dönüştürülebilir. d' örnek varyansı ve gerçek varyans arasındaki farkı ifade etmektedir¹²⁴.

Örnek hacminin belirlenmesinde elbette diğer yaklaşımlar mevcuttur. Simülasyon sonuçları kullanılarak güvenlik sınırı saptanabilir. Her bir denemeden sonra ortalama değeri hesaplamak için diğer bir yaklaşım aşağıdaki şekilde görüldüğü gibi bir limite doğru gitmektir. En yüksek ve en düşük noktalar arası mesafe, bir değere eşit veya ondan daha küçük olduğu zaman simülasyon durdurulur. Bu yöntem standart sapma ve varyans için de kullanılabilir¹²⁵.

¹²⁴ Halaç, s: 469

¹²⁵ Halaç, s: 469



Şekil 10. Stokastik Limit

4.6. Rassal Sayı Üreteçlerinin Testi

Verilen bir sayı dizisi üzerinde yapılacak olan çeşitli testler, sayı dizisinin ‘rasgele’ tarifine uygun olup olmadığını kontrol etmek için yapılır. Fakat bu testler hiçbir zaman mutlak uygunluğu ispat etmez. Verilen testleri geçen bir dizi üzerinde test edilmemiş ve rasgele olmayan bir özellik bulunabilir. Bu nedenle sadece testlerin red sonucu anlamlıdır. Bununla beraber bir grup testi geçen sayılar Monte Carlo deneylerinde kullanılır. Özel amaçlarla kullanılacak olan rasgele sayılar için amaca uygun testler yapılmalıdır.

Sayılar üzerinde yapılacak olan testleri deneysel ve teorik olmak üzere ikiye ayırmak gereklidir. Deneysel testler rasgele sayılar meydana getiren algoritmaların neticeleri üzerinde yapılan testlerdir. Sayıları meydana getiren algoritma teorik olarak incelenerek söz konusu algoritma ile meydana getirilen sayıların rasgelelik tanımına uyacağı kestirilebilir¹²⁶.

¹²⁶ Yücel, s: 51 - 52

4.6.1. X^2 Testi

Bu testle, elde edilen rassal sayıların $(R_i, i=1,2,\dots)$ $(0,1)$ aralığında düzgün dağılıp dağılmadığını test ederiz. Test, aynı zamanda bu rassal sayıların bağımsız olarak dağıldığını varsaymaktadır.

Rassal sayı üreticinden ardışık olan n adet sayı alınır. $(0,1)$ aralığı k adet alt aralığa bölünür.

$$(0,h), [h,2h), \dots, [(k-1)h,kh), kh=1$$

H_0 hipotezi altında elde edilen rassal sayıların $U(0,1)$ dağılımından geldiği test edilir.

H_0 hipotezi altında test istatistiği

$$\chi^2 = \sum_{i=1}^k \frac{(G_i - B_i)^2}{B_i} = \sum_{i=1}^k \frac{G_i^2}{B_i} - n \quad (G_i \text{ gözlenen frekans, } B_i \text{ beklenen}$$

frekanstır)

$k-1$ serbestlik dereceli X^2 dağılımıdır. Hesaplanan istatistik değeri α anlam düzeyi ve $k-1$ serbestlik dereceli tablo değerinden büyükse sıfır hipotezi reddedilir; örneğin düzgün dağıldığını ispatlayacak yeterli delil bulunamamıştır kararı verilir¹²⁷.

Örnek olarak rassal sayı üreticilerinden biri kullanılarak 1000 rassal sayı elde edildiğini ve $h=0.1$ olacak şekilde aşağıdaki gibi $k=10$ alt aralığa ayrıldığını düşünelim.

Tablo 8
 X^2 Testi Örneği

Aralıklar	[0,h)	[h,2h)	[2h,3h)	[3h,4h)	[4h,5h)	[5h,6h)
G_i	99	94	95	108	108	88
B_i	100	100	100	100	100	100
	[6h,7h)	[7h,8h)	[8h,9h)	[9h,10h)		
	111	92	111	94		
	100	100	100	100		

¹²⁷ Dagpunar, s: 29

$$\chi^2 = \frac{100676}{100} - 1000 = 6,76 \text{ olarak hesaplanır. Hesaplanan test istatistik değeri,}$$

tablo değeri olan $\chi^2_{9,0,05} = 16,92$ değerinden küçük olduğundan (Aynı zamanda gözlem değerlerinin bağımsız olduğunu varsayan) H_0 hipotezini reddedecek yeterli kanıt bulunamamıştır; gözlem değerleri $U(0,1)$ dağılımından gelmektedir. (Beklenen frekans değerlerinin 5'e eşit veya büyük olması gerekmektedir)

Ki Kare testinin dezavantajı öncelikle gözlemlerin bağımsızlığının test edilmesi gerekliliğidir. Daha sonra gözlem değerlerinin alt aralıklara ayrılması ve sürekli dağılımlardan ziyade kesikli dağılımlar için $U(0,1)$ dağılımına uygunluğun aranmasıdır¹²⁸.

4.6.2. Kolmogorov – Smirnov Testi

Kolmogorov – Smirnov testinde, dağılım fonksiyonu esas olarak alınır. Bir x rasgele değişkeninin dağılım fonksiyonu $F(x)$ ise bu değişken üzerinde n adet gözlem yapılmışsa

$$nF_n(x) = \{x' \text{ ten küçük olan gözlemlerin sayısı} \}$$

eşitliği ile tanımlanan $F_n(x)$ deneysel dağılımı ile $F(x)$ teorik dağılımı karşılaştırılır. Bu amaç için

$$k = \sqrt{n} \max_x |F_n(x) - F(x)|$$

sapması hesaplanarak χ^2 testinde olduğu gibi özel tablolardaki değerlerden yüksek olup olmadıkları kontrol edilir. $F_n(x)$ fonksiyonunu bulmak için gözlemlerin büyüklük sırasına dizilmesi gerekir. Bu işlem için verimli algoritmalar vardır.

Rasgele sayı dizilerinde verilen $[0,0.1)$, $[0.1,0.2)$, ..., $[0.9,1)$ aralıklardaki rakamlar sayılır ve teorik frekansları olan $1/10$ ile karşılaştırılır¹²⁹.

¹²⁸ Dagpunar, s: 30

¹²⁹ Yücel, s: 53

4.6.3. Çiftler Testi

(X_i, X_{i+1}) rasgele sayı çifti ile bir düzlem koordinat sistemindeki bir birim karede tanımlanan noktanın dağılımı düzgün olmalıdır. Aralık N eşit aralığa bölünürse alan N^2 'ye bölünmüş olur. Her bir karedeki noktaların sayısının teorik sayıdan sapması için yine bir χ^2 testi yapılabilir.

Rasgele sayı dizilerindeki bir i, j rakam çiftinin bu sıra ile ve ardışık olarak bulunması ihtimali $1/100$ 'dür. Bu teorik sapma X^2 ve Kolmogorov – Smirnov testlerinde olduğu gibi kontrol edilir¹³⁰.

¹³⁰ Yücel, s: 53

5. TEK KANALLI KUYRUK MODELİNDE MONTE CARLO SİMÜLASYON METODUNUN BİR İŞLETMEDE UYGULANMASI

5.1. Uygulamanın Yapıldığı İşyeri İle İlgili Bilgiler

Bu çalışma için endüstrinin ve otomotiv sektörünün geliştiği kentlerden biri olan Kocaeli'ndeki küçük çaplı bir işletme seçilmiştir. Dört çalışanı bulunan bu işletmede çalışma şekli vardiya usulüne dayanmakta ve gündüz 09:00 ile akşam 18:00 saatleri arasında olmak üzere bir vardiya sistemi kullanılmaktadır. Çalışmada kullanılan veriler 07.04.2008 ve 11.04.2008 tarihleri arasında, beş gün süreyle işyerinde gözlem yapılarak elde edilmiştir.

İşyerinin faaliyet konusu genel olarak soğuk demir işçiliği, hidrolik ve pnömatik ekipmanların ve aparatların imalatı, bakımı ve tamiri olarak gerçekleşmektedir.

5.2. Uygulamanın Amacı

Bu uygulamanın amacı;

- 1) Bir günde gelen ortalama müşteri sayısının tespiti,
- 2) Müşterilerin geliş dağılımlarının saptanması,
- 3) Ortalama servis zamanının saptanması,
- 4) Servis süresinin dağılımının belirlenmesi,
- 5) Müşterilerin kuyrukta bekleme zamanının belirlenmesi,
- 6) Kuyruk uzunluğunun belirlenmesi,
- 7) Bir müşterinin ortalama sistemde geçirdiği zamanın belirlenmesi,
- 8) Servis sağlayıcının boş kaldığı sürelerin belirlenmesi,
- 9) Sistemin kapasite kullanım oranının belirlenmesi,

10) Elde edilen bilgilerle sistemin geliştirilmeye çalışılmasıdır.

Uygulama çerçevesinde bu veriler daha sonraki dönemlerde işyeri çalışanları tarafından kullanılabilir.

5.3. Araştırmada Kullanılan Yöntem

Araştırmada kullanılan veriler bu işyerinde beş gün süren gözlemler sonunda elde edilmiştir. İşyerinde yapılan gözlem sonucu elde edilen verilere ait çizelge oluşturulmuş ve bu çizelge MS Office uygulamalarından Excel ile doküman haline getirilmiştir. Uygun formüllerin girilmesi ile uygulama çalıştırılmış ve parametre tahminleri yapılmıştır.

5.4. Sistemin Tanımlanması

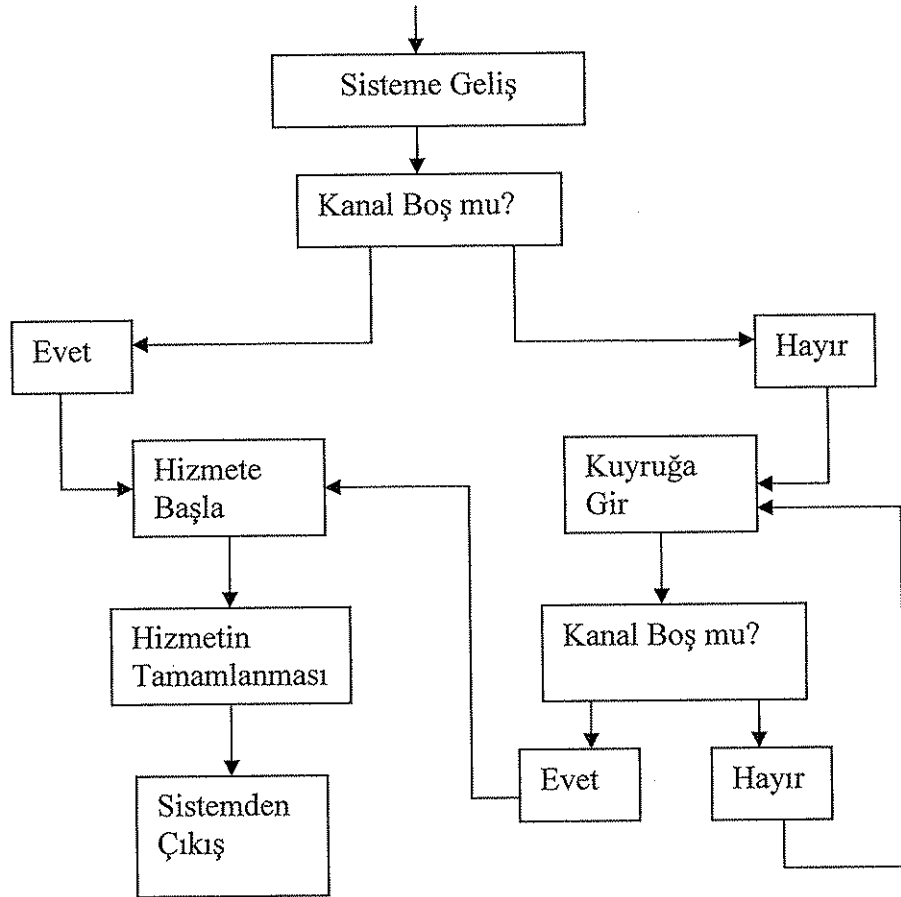
Uygulamada sistemin tek kanallı tek servis sunuculu M/M/1 modeline uyduğu düşünülmektedir. Burada servis süresinin deterministik olduğu düşünülebilir; halbuki servis süresi tam olarak deterministik değildir. Çünkü servisi sağlayan işçidir ve tam olarak otomasyona geçildiğinden söz edilemez. Eğer bu servis bir üretim bandında gerçekleşiyor olsaydı servis sürelerinin deterministik olduğundan bahsedebilirdik.

Servis için gelen müşteriler hemen hemen birbirinin aynısı olan parçaların imalatı veya bakımı için geldiklerinden servis süreleri de farklı olmaktadır. Elde edilen verilerden de servis süresinin sabit olmadığı, dolayısıyla deterministik olamayacağından söz edebiliriz. (Deterministik – Sabit servis süreli modellerin uygulanması için de M/M/1 modeli kullanılır. Araştırmanın daha önceki bölümlerinde M/D/1 sistemlerinde performans ölçülerinin M/M/1 modellerinin performans ölçülerinin yarısı olduğundan bahsedilmişti.)

Gelen müşteriler, kanalın boş olması halinde servise alınmaktadır. Kanal dolu ise servis sağlayıcısının boşalmasını beklemek için kuyruğa girerler. Belli bir süre kuyrukta geçirip daha sonra boşalan kanala geçerek hizmet görürler. Hizmet sırasında öncelik sistemi işlememektedir.

Müşterilerin işyerine gelmesi ve taleplerini iletmesinin ardından servisi sağlayan işçi eldeki parçanın boyutlarını ölçmekte, planı incelemekte ve parçaları CNC model tezgaha yerleştirerek servisi başlatmaktadır.

Bu sistemin işleyişini aşağıdaki şekilde gösterebiliriz:



Şekil 11. Kuyruk Sisteminin Şematik Olarak Gösterimi

5.5. Verilerin Kullanılması

Uygulamada gözlenen veriler sayısal değer ifade edebilmesi açısından dakika/saat cinsinden hesaplanmış ve bu şekilde tablolarda kullanılmışlardır.

Gözlem değerleri, kuyruk teorisinin parametrelerini tahmin etmeye yarayacak olan verilerdir.

Çalışmada şu veriler elde edilmiştir;

1) Müşterilerin sisteme geliş anı: Saat cinsinden gözlemlenmiş fakat direkt olarak kullanılmamıştır. Dakikaya çevrilerek sayısal anlam kazandırılmıştır.

2) Müşterilerin sisteme gelişleri arasındaki zaman: İki müşteri arasında sisteme geliş süreleri hesaplanmıştır. Bu hesap ayrıca gelişler arası sürenin Poisson olup olmadığının testinde de kullanılacaktır. Çünkü eğer gelişler arası süre gerçekten Poisson dağılımına uyuyorsa simülasyon metodunu kullanmaya gerek kalmayacak, M/M/1 kuyruk modeliyle çalışmaya devam edilecektir.

3) Servis süresi: Servis başlangıç ve servis süresi hesaplanmıştır. Müşteri gelişleri arasındaki sürede olduğu gibi servis süresinin de Üssel dağılıma uyduğu yapılan test sonucu kabul edilirse simülasyon yöntemine gerek kalmayacaktır.

4) Müşterilerin sistemde geçirdiği ortalama süre: Bu süre sistem performansının ölçülmesi için gereklidir. Veriler kullanılarak müşterilerin kuyrukta ve servis süresince geçirdikleri zaman hesaplanmıştır.

Beş günlük sürede gözlemlenen veriler aşağıdadır.

Tablo 9
Müşterilerin Gelişler Arası Süresi, Sistemde Geçirdikleri ve
Servise Başladıkları Süreler

Müşteri	Gelişler Arası Süre	Geliş Anı	Servis Başlangıcı
1	0,45	0,45	0,45
2	2,30	2,75	2,75
3	1,83	4,58	4,58
4	1,08	5,66	6,78
5	1,75	7,41	8,05
6	1,09	8,50	8,96
7	1,91	10,41	10,41
8	1,58	11,99	11,99
9	0,63	12,62	13,52
10	1,96	14,58	15,00
11	1,48	16,06	16,96
12	0,75	16,81	18,21
13	0,50	17,31	18,79
14	3,21	20,52	20,52
15	0,75	21,27	21,52
16	1,58	22,85	23,69
17	1,53	24,38	24,38
18	1,68	26,06	26,06
19	1,28	27,34	28,36
20	1,58	28,92	29,77
21	1,58	30,50	32,02
22	2,25	32,75	33,77
23	1,91	34,66	35,73
24	1,08	35,74	38,14
25	1,71	37,45	39,29
26	1,28	38,73	40,12
27	1,61	40,34	42,34
28	1,38	41,72	44,11
29	1,41	43,13	46,52
30	2,41	45,54	48,70

İşyerinin çalışma saatleri sabah 09:00 ve akşam 18:00 saatleri arasındadır. İlk müşteri sisteme 09:27’de gelmektedir. 09:00, sistemin çalışmaya başladığı saattir ve 27. dakikada ilk müşterinin sisteme geldiği ve sunucu boş olduğu için servise alındığı zamandır. Diğer bir deyişle, $27/60=0,45$ ’te ilk müşteri sisteme dahil olmakta ve servise başlamaktadır. İkinci müşteri sisteme saat 11:45’te gelmekte, 09:00 ile arasındaki zaman farkı olan 2 saat 45 dakika (165 dakika) $165/60=2,75$ şeklinde sisteme geliş anı

olarak kaydedilmektedir. İlk müşteri ile arasında 138 dakika bulunmaktadır. Bu süre de $138/60=2,30$ olarak kaydedilmiştir. Benzer şekilde servise başlama süresi, servis süresi, servisin bittiği süre, sunucunun boş kaldığı süre, müşterinin sistemde geçirdiği ve kuyrukta beklediği süreler de hesaplanmıştır.

Tablo 10
Müşteriler İçin Servis süresi, Servisin Bitişi ve
Sunucunun Boş Kaldığı Süreler

Müşteri	Servis Süresi	Servisin Bitişi	Sunucunun Boş Kaldığı Süre
1	2,25	2,70	0,45
2	1,55	4,30	0,05
3	2,20	6,78	0,28
4	1,27	8,05	0,00
5	0,91	8,96	0,00
6	1,33	10,29	0,00
7	1,09	11,50	0,12
8	1,53	13,52	0,49
9	1,48	15,00	0,00
10	1,96	16,96	0,00
11	1,25	18,21	0,00
12	0,58	18,79	0,00
13	0,33	19,12	0,00
14	1,00	21,52	1,40
15	2,17	23,69	0,00
16	0,66	24,35	0,00
17	1,57	25,95	0,03
18	2,30	28,36	0,11
19	1,41	29,77	0,00
20	2,25	32,02	0,00
21	1,75	33,77	0,00
22	1,96	35,73	0,00
23	2,41	38,14	0,00
24	1,15	39,29	0,00
25	0,83	40,12	0,00
26	2,22	42,34	0,00
27	1,77	44,11	0,00
28	2,41	46,52	0,00
29	2,18	48,70	0,00
30	1,16	49,86	0,00

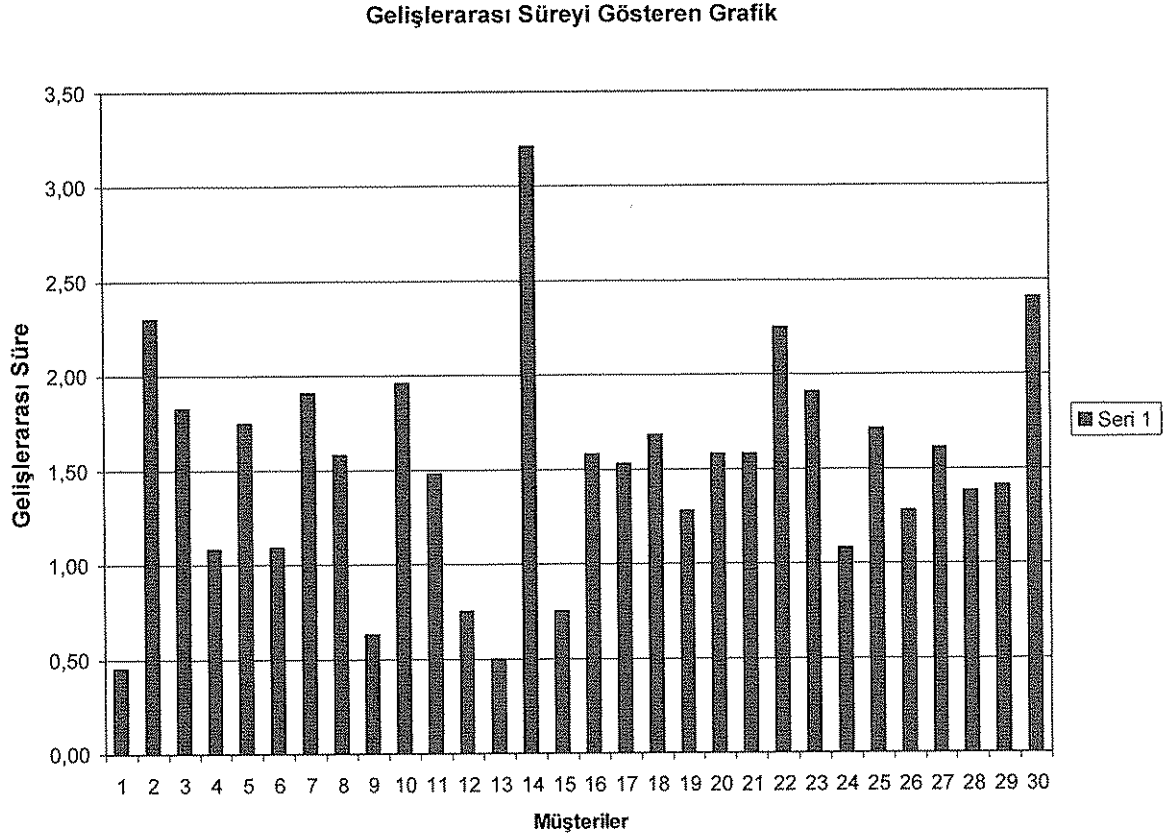
Gelişler arası sürede olduğu gibi servis sürelerinin de belirli bir dağılımdan gelmesi uygulamada bazı kolaylıklar sağlayabilir. Servis sürelerinin üssel dağılıma uyması kuyruk modelleriyle çalışabilme imkanı verecektir. Bunun için önce servis sürelerinin bu dağılıma uygunluğunun test edilmesi gerekmektedir.

Tablo 11

Müşterilerin Sistemde Geçirdiği Süre, Bekleme Süresi, Kuyruktaki ve Sistemdeki Müşteri Sayıları

Müşteri	Müşterinin Sistemde Geçirdiği Süre	Bekleme Süresi	Kuyruktaki Müşteri Sayısı	Sistemdeki Müşteri Sayısı
1	2,25	0,00	0	1
2	1,55	0,00	0	1
3	2,20	0,00	0	1
4	2,39	1,12	1	2
5	1,55	0,64	1	2
6	1,79	0,46	1	2
7	1,09	0,00	0	1
8	1,53	0,00	0	1
9	2,38	0,90	1	2
10	2,38	0,42	2	3
11	2,15	0,90	2	3
12	1,98	1,40	1	2
13	1,81	1,48	1	2
14	1,00	0,00	0	1
15	2,42	0,25	1	2
16	1,50	0,84	1	2
17	1,57	0,00	0	1
18	2,30	0,00	0	1
19	2,43	1,02	1	2
20	3,10	0,85	1	2
21	3,27	1,52	1	2
22	2,98	1,02	1	2
23	3,48	1,07	1	2
24	3,55	2,40	1	2
25	2,67	1,84	1	2
26	3,61	1,39	1	2
27	3,77	2,00	1	2
28	4,80	2,39	1	2
29	5,57	3,39	1	2
30	4,32	3,16	1	2

Müşterilerin kuyrukta ve sistemde geçirdiği süreler uygulama sonucu işletme sahibine fikir verir. Sisteme yeni kanal ilave etmek ve ilave edilen yeni kanalın maliyetinin hesaplanması buna örnek olarak verilebilir.

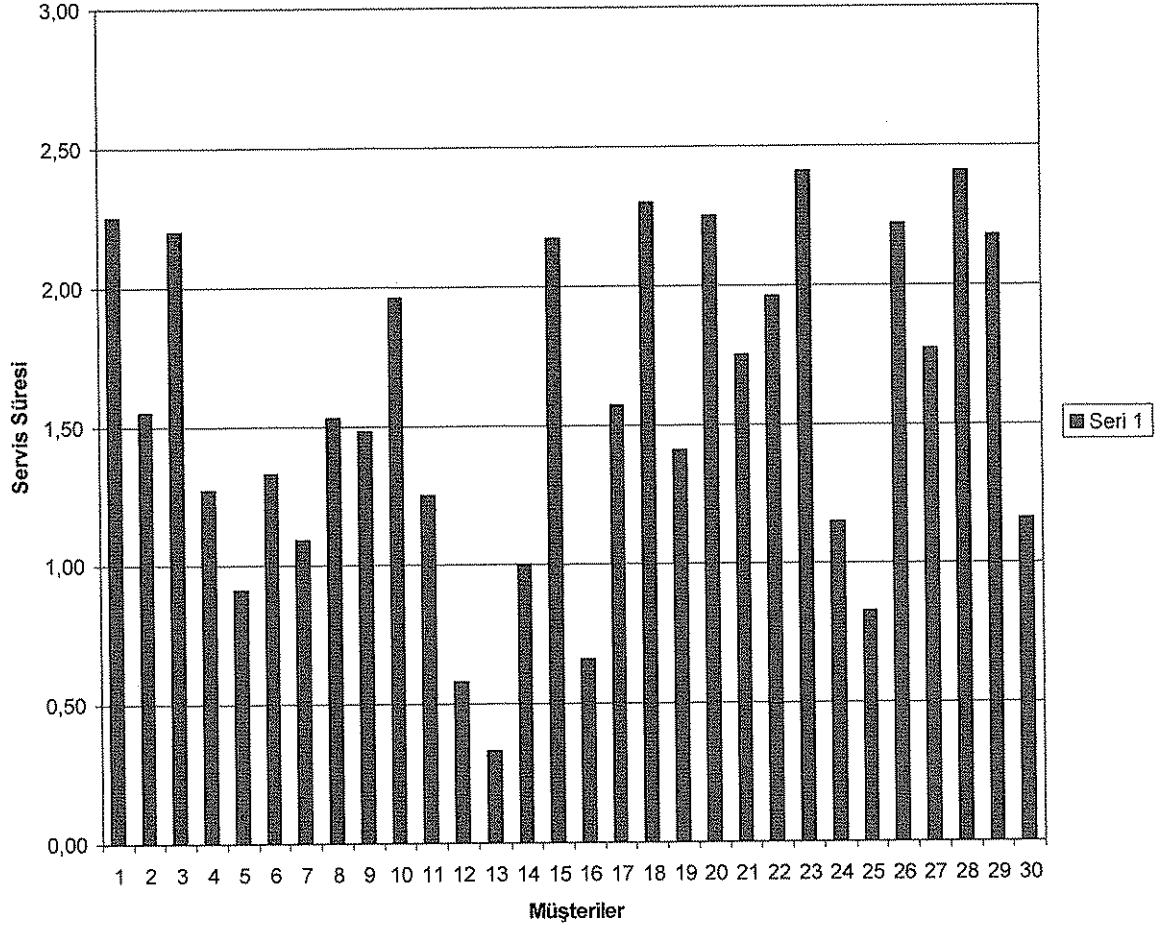


Şekil 12. Müşterilerin Sisteme Gelişler Arası Süresini Gösteren Grafik

Birinci müşteri sisteme 09:27'de gelmektedir. En küçük değer de 0,45 ile birinci müşteriye ait olduğu görülür. En büyük değer ise 3,21 ile 14. müşteriye aittir. 14. müşteri sisteme 13. müşteriden 195 dakika sonra gelmektedir.

Gelişler arası sürenin hesaplanması uygulama için önem taşımaktadır. Şayet; gelişler arası sürenin dağılımı Poisson dağılımına uyarsa kuyruk modellerinden bir tanesi ile çalışma olanağı doğacak, dolayısıyla simülasyon metoduna gerek kalmayacaktır.

Servis Sürelerini Gsteren Grafik

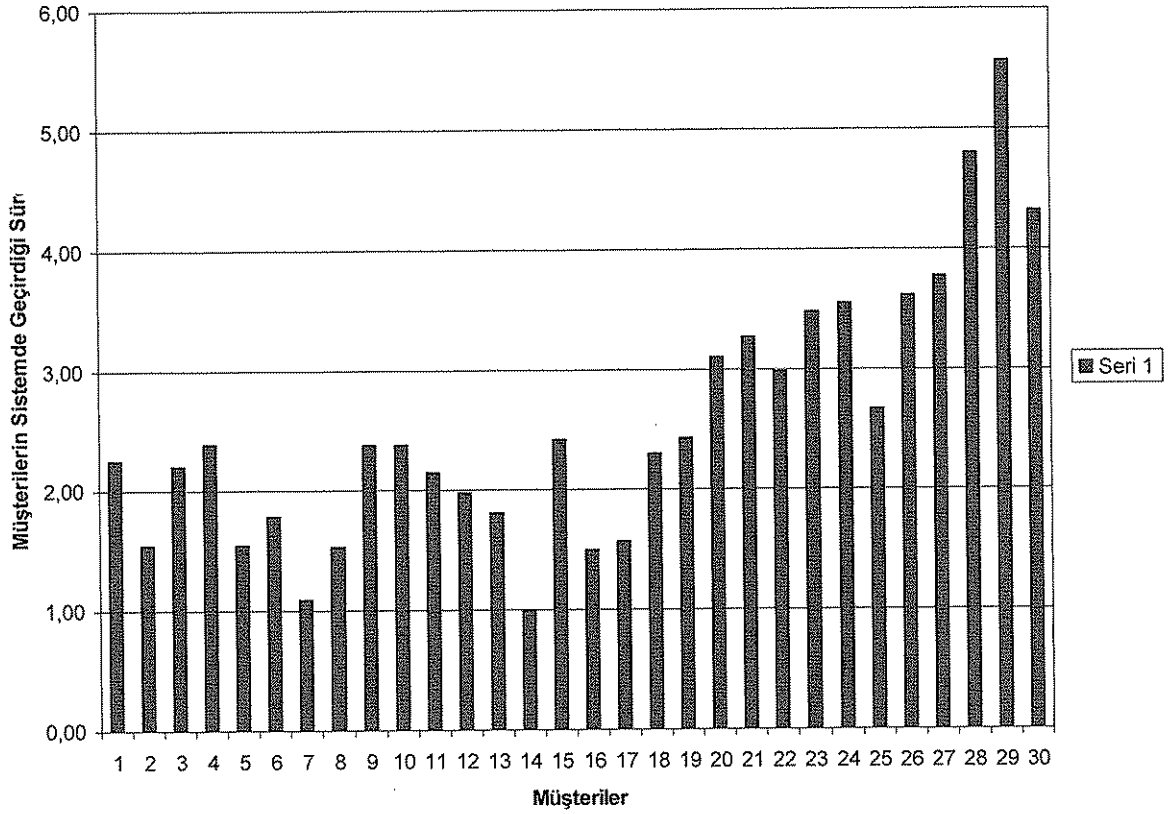


Şekil 13. Servis Sürelerini Gösteren Grafik.

En kısa servisi 13. müşteri almıştır. 13. müşterinin servis süresi 20 dakika, karşılık geldiği değer de 0,33'tür. En uzun süre servis alan müşteriler de 23. ve 28. müşterilerdir. Her iki müşterinin de servis süresi 145 dakikadır.

Servis sürelerinin hesaplanması da gelişler arası sürenin hesaplanması kadar önemlidir. Çünkü gelişler arası sürenin Poisson, servis sürelerinin de Üssel dağılıma uygunluğu sağlanırsa yukarıda da belirtildiği gibi simülasyon metodu yerine kuyruk teorisi ile çalışmaya devam edilecektir.

Müşterilerin Sistemde Geçirdiği Süreyi Gösteren Grafik



Şekil 14. Müşterilerin Sistemde Geçirdiği Süreler

Sistemin çalışmaya başladığı ilk anlarda müşterilerin sistemde daha kısa süre kaldıkları görülmektedir. Sistemin son zamanlarında ise servis süresinin uzaması sebebiyle kuyruklar oluşmakta ve müşteriler sistemde daha uzun süre geçirmesi gerekmektedir. Sistemde en uzun süre kalan müşteri 335 dakika (5 saat 35 dakika) ile 29. müşteridir. (Gerçek uygulamada bu işyerine gelen müşteriler bu süreyi işyerinde bekleyerek geçirmemektedir. İmalat veya bakım için gelen müşteriler, ellerindeki planı ve parçaları işyerine bırakmakta ve bittikleri zaman gelip almaktadırlar. Bu süre teorik olarak kullanılmaktadır.)

5.6. Gelişler Arası Sürelerin Poisson Dağılımına Uygunluğunun Testi

Sıra bekleme sistemlerinin incelenmesinde gelişlerin dağılımı belirlendikten sonra akla ilk gelen şey, bu dağılımın Poisson olup olmadığıdır. Çünkü eğer gelişler

Poisson dağılımı ise mevcut kuyruk teorisi modellerinden birinin kullanılabilme olanağı ortaya çıkabilir. Dolayısıyla bir simülasyon modeli geliştirmeye de gerek kalmayabilir. Kuyruk teorisi modellerinden birinin kullanılabilme olanağı doğmasa bile gelişlerin Poisson dağılım göstermesi simülasyon modellerinin kurulmasında ve deneylerin yapılmasında büyük kolaylık sağlar¹¹⁸.

Geliş dağılımının Poisson olup olmadığını değişim katsayısını kullanarak anlayabiliriz.

$$\text{DeğişimKatsayısı} = \frac{\text{Standart Sapma}(s)}{\text{Aritmetik Ortalama}(\lambda)} = \frac{1}{\sqrt{\lambda}} \text{ 'dır.}$$

Bu eşitlik sağlandığında geliş dağılımlarının Poisson dağılımına uyduğunu söyleyebiliriz.

Gelişler arası süre 1,52 saat ve varyansı da 0,36 ise değişim katsayısı;

$$\text{Değişim Katsayısı} = \frac{\sqrt{0,36}}{1,52} = \frac{1}{\sqrt{1,52}} \\ 0,3947 \neq 0,8111$$

olduğundan gelişler arası sürenin dağılımı Poisson dağılımına uymamaktadır.

Bu farkın istatistiksel olarak da test edilmesi gerekir. Bunun için χ^2 test istatistiği oluşturulup gelişler arası süreler bu teste tabi tutulacaktır. ($\alpha = 0,95$)

χ^2 testini aşağıdaki şekilde uygularız;

H_0 : Gelişler arası sürelerin dağılımı Poisson dağılımına uymaktadır.

H_1 : Gelişler arası sürelerin dağılımı Poisson dağılımına uymamaktadır.

¹¹⁸ Sariaslan, s : 57

Tablo 12

 χ^2 Testinde Kullanılacak Gözlenen Frekans Değerleri

Gelişler Arası Süre	Geliş Sayıları	Geliş Sayıları	Gözlenen Frekans
0-0,60	2	1	2
0,60 – 1,20	6	2	1
1,20 – 1,80	14	6	2
1,80 – 2,40	6	14	1
2,40 – 3,00	1		
3,00 – 3,60	1		
Toplam	30		

χ^2 testinin uygulanabilmesi için hiç bir aralıkta 5'ten az frekansın bulunmaması gerekmektedir. Ayrıca χ^2 testinin gücü, serbestlik derecesinin yüksek olmasına dayanır. Uygulamamızda örnek sayısını az olmasından dolayı gözlenen frekansların hücre değerlerinin tamamı 5'ten küçüktür. Dolayısıyla bu uygulamada χ^2 test istatistik değeri oluşturulamayacaktır.

Ampirik dağılımların, özellikleri bilinen standart istatistiksel dağılımlara uygunluğunu belirlemek için χ^2 testinden başka yöntemler de kullanılabilir. χ^2 testi bu amaçla en sık kullanılan yöntem olmasına karşılık Kolmogorov – Smirnov, Anderson – Darling ve Cramer Von Mises testleri de sık sık kullanılır¹¹⁹.

Kolmogorov – Smirnov testinin χ^2 testine üstünlüğü gözlenen frekansların 5'ten büyük olma koşulunu getirmemesi ve az sayıdaki aralıkla çalışıldığında da güvenilir sonuçlar vermesidir.

Uygulamaya χ^2 testinde olduğu gibi hipotezlerin kurulmasıyla başlanır. (Herhangi bir dağılımın belli bir dağılıma uygunluğunu araştırırken oluşturulacak hipotezler değişmez. Anderson – Darling testi için oluşturulacak hipotezler de aynıdır.) ($\alpha = 0,95$)

¹¹⁹ Sariaslan, s : 67

H_0 : Gelişler arası sürelerin dağılımı Poisson dağılımına uymaktadır.

H_1 : Gelişler arası sürelerin dağılımı Poisson dağılımına uymamaktadır.

Testin uygunluğu MS Office Excel uygulamasının EasyFit 4. 3 eklentisiyle yapılmıştır. Buna göre oluşturulan tablolara ait sonuçlar aşağıdadır.

Tablo 13.

Kolmogorov – Smirnov ve Anderson Darling Testi Sonuçları.

Goodness of Fit - Summary

#	Distribution	Kolmogorov Smirnov		Anderson Darling	
		Statistic	Rank	Statistic	Rank
1	Poisson	0,306466433	1	50,8299406	1

Goodness of Fit - Details [hide]

Poisson [#1]

Kolmogorov-Smirnov

Sample Size	30				
Statistic	0,306466433				
P-Value	0,005344956				
Rank	1				
α	0,2	0,1	0,05	0,02	0,01
Critical Value	0,19032	0,21756	0,2417	0,27023	0,28987
Reject?	Yes	Yes	Yes	Yes	Yes

Anderson-Darling

Sample Size	30				
Statistic	50,8299406				
Rank	1				
α	0,2	0,1	0,05	0,02	0,01
Critical Value	1,37490912	1,9286236	2,50175596	3,28919661	3,90741594
Reject?	Yes	Yes	Yes	Yes	Yes

Kolmogorov – Smirnov testinde test istatistik değeri 0,3064 olarak bulunmuştur. Bu değer 0,05 hata payı ile tablo değeri olan 0,2417’den büyük olduğu için (Diğer bir deyişle kuyruk olasılık değeri olan 0,0053 ; 0,05’ten küçük olduğu için) H_0 hipotezinin kabul edilemeyeceği yönünde karar verilir. Gelişler arası sürelerin dağılımı Kolmogorov – Smirnov testi sonucunda Poisson dağılımına uymadığı söylenir.

Anderson Darling testi için de aynı sonuç görülmektedir. Test istatistik sonucu olan 50,8299 , 0,05 hata payına göre kritik değer olan 2,5017’den büyük olduğu için H_0 hipotezi reddedilir.

5.7. Servis Sürelerinin Üssel Dağılıma Uygunluğunun Testi

Gelişler arası sürelerin dağılımının yapılan her üç testte de Poisson dağılımına uymadığı görülmektedir. Aynı şekilde servis sürelerinin üssel dağılıma uyup uymadığı araştırılacaktır. Bunun için de aynı yöntem kullanılır.

H_0 : Servis sürelerin dağılımı Üssel dağılıma uymaktadır.

H_1 : Servis sürelerin dağılımı Üssel dağılıma uymamaktadır.

Tablo 14

Servis Sürelerinin Üssel Dağılıma Uygunluğu – Test Sonuçları

Goodness of Fit - Summary							
#	Distribution	Kolmogorov Smirnov		Anderson Darling		Chi-Squared	
		Statistic	Rank	Statistic	Rank	Statistic	Rank
1	Erlang	0,226153113	4	3,05851152	4	6,79630194	4
2	Erlang (3P)	0,148124782	2	0,509236593	2	0,103165606	1
3	Exponential	0,311734826	6	5,10325222	6	9,74001924	6
4	Exponential (2P)	0,259746598	5	4,26607272	5	7,10995252	5
5	Gamma	0,148950955	3	0,773583622	3	0,13686177	3
6	Gamma (3P)	0,141279179	1	0,471916213	1	0,110178907	2

Goodness of Fit - Details [hide]**Exponential** [#3]**Kolmogorov-Smirnov**

Sample Size	30				
Statistic	0,311734826				
P-Value	0,004347247				
Rank	6				
α	0,2	0,1	0,05	0,02	0,01
Critical Value	0,19032	0,21756	0,2417	0,27023	0,28987
Reject?	Yes	Yes	Yes	Yes	Yes

Anderson-Darling

Sample Size	30				
Statistic	5,10325222				
Rank	6				
α	0,2	0,1	0,05	0,02	0,01
Critical Value	1,37490912	1,9286236	2,50175596	3,28919661	3,90741594
Reject?	Yes	Yes	Yes	Yes	Yes

Chi-Squared

Deg. of freedom	2				
Statistic	9,74001924				
P-Value	0,007673291				
Rank	6				
α	0,2	0,1	0,05	0,02	0,01
Critical Value	3,21887582	4,60517019	5,99146455	7,82404601	9,21034037
Reject?	Yes	Yes	Yes	Yes	Yes

Kolmogorov – Smirnov testinde test istatistik değeri olan 0,3117 , 0,05 hata payına göre kritik değeri olan 0,2417’den büyük olduğundan (Kuyruk olasılık değeri olan 0,0043; 0,05’ten küçük olduğu için) H_0 hipotezi reddedilir. Servis sürelerinin dağılımı üssel dağılıma uymamaktadır.

χ^2 testi sonucunda da aynı kaniya varılır. Test istatistik değeri olan 9,7400 , 0,05 hata payıyla kritik değeri olan 5,99’den büyük olduğundan (Kuyruk olasılık değeri

olan 0,0076; 0,05'ten küçük olduğu için) H_0 hipotezi reddedilir. Kolmogorov – Smirnov testinde olduğu gibi χ^2 testinde de servis sürelerinin dağılımının üssel dağılıma uymadığı görülür.

Bu sonuç, teorik olarak M/M/1 kuyruk modeliyle çalışma olanağımızın olmadığını gösterir. Eğer mevcut koşullar altında kuyruk teorisinin kullanılması zorunluysa gelişler arası sürenin ve servis sürelerinin hangi dağılımdan geldiği araştırılmalı ve uygun model oluşturulmalıdır. Çoğu uygulama için Poisson dağılımına uymayan geliş sürelerini dikkate alarak farklı modeller geliştirilmiştir.

5.8. Birikimli Olasılık Fonksiyonlarının Oluşturulması Ve Monte Carlo Simülasyon Metodunun Uygulaması

Monte Carlo metodunun, varsayımların aksadığı durumlarda, teorik olarak uygulanmasında sakınca olan, kesin ve güvenilir sonuçların elde edilemeyeceği uygulamalar için kullanılabileceğinden bahsetmiştik. Bu metodun kesikli olay simülasyonlarında en çok uygulandığı alanlardan biri de kuyruk modelleridir.

Yöntemin uygulanmasına gelişler arası süreler ve servis sürelerinin birikimli olasılık değerlerini oluşturarak başlarız. Bunun için gelişler arası süreler ve servis süreleri için frekans değerleri kullanılır. Frekans değerleri yapılan gözlemden elde edilmiştir.

Tablo 15.

Gelişler Arası Süreler İçin Rassal Sayı Aralıklarının Oluşturulması

Gelişler Arası Süre	Geliş Sayısı	Geliş Olasılıkları	Birikimli Olasılıklar	Rassal Sayı Aralığı	Rassal Sayı *
0,45	1	0,0333	0,0333	0 – 0,0332	0,9296
0,50	1	0,0333	0,0666	0,0333 – 0,0665	0,8902
0,63	1	0,0333	0,0999	0,0666 – 0,0998	0,3164
0,75	2	0,0666	0,1665	0,0999 – 0,1664	0,1307
1,08	2	0,0666	0,2331	0,1665 – 0,2330	0,1839
1,09	1	0,0333	0,2664	0,2331 – 0,2663	0,0398
1,28	2	0,0666	0,3330	0,2664 – 0,3329	0,2046
1,38	1	0,0333	0,3663	0,3330 – 0,3662	0,8264
1,41	1	0,0333	0,3996	0,3662 – 0,3995	0,5677
1,48	1	0,0333	0,4329	0,3996 – 0,4328	0,5321
1,53	1	0,0333	0,4662	0,4329 – 0,4661	0,5955
1,58	4	0,1332	0,5994	0,4662 – 0,5993	0,9563
1,61	1	0,0333	0,6327	0,5994 – 0,6326	0,9645
1,68	1	0,0333	0,6660	0,6327 – 0,6659	0,4619
1,71	1	0,0333	0,6993	0,6660 – 0,6992	0,6532
1,75	1	0,0333	0,7326	0,6993 – 0,7325	0,9237
1,83	1	0,0333	0,7659	0,7326 – 0,7658	0,7489
1,91	2	0,0666	0,8325	0,7659 – 0,8324	0,3739
1,96	1	0,0333	0,8658	0,8325 – 0,8657	0,6536
2,25	1	0,0333	0,8991	0,8658 – 0,8990	0,1550
2,30	1	0,0333	0,9324	0,8991 – 0,9323	0,7477
2,41	1	0,0333	0,9657	0,9324 – 0,9656	0,8923
3,21	1	0,0333	1	0,9657 – 0,9999	0,9613

* MS Excel uygulamasıyla rassal sayılar elde edilmiştir.

Tablo 16.
Servis Süreleri İçin Rassal Sayı Aralıklarının Oluşturulması

Servis Süresi	Servis Sayısı	Olasılık	Birikimli Olasılık	Rassal Sayı Aralığı	Rassal Sayı *
0,33	1	0,0333	0,0333	0 – 0,0332	0,6606
0,58	1	0,0333	0,0666	0,0333 – 0,0665	0,2001
0,66	1	0,0333	0,0999	0,0666 – 0,0998	0,4136
0,83	1	0,0333	0,1332	0,0999 – 0,1331	0,8136
0,91	1	0,0333	0,1665	0,1332 – 0,1664	0,6681
1,00	1	0,0333	0,1998	0,1665 – 0,1997	0,4032
1,09	1	0,0333	0,2331	0,1998 – 0,2330	0,2299
1,15	1	0,0333	0,2664	0,2331 – 0,2663	0,6878
1,16	1	0,0333	0,2997	0,2664 – 0,2996	0,8077
1,25	1	0,0333	0,3330	0,2997 – 0,3329	0,2371
1,27	1	0,0333	0,3663	0,3330 – 0,3662	0,6368
1,33	1	0,0333	0,3996	0,3663 – 0,3995	0,4549
1,41	1	0,0333	0,4329	0,3996 – 0,4328	0,1722
1,48	1	0,0333	0,4662	0,4329 – 0,4661	0,5997
1,53	1	0,0333	0,4995	0,4662 – 0,4994	0,2604
1,55	1	0,0333	0,5328	0,4995 – 0,5327	0,6349
1,57	1	0,0333	0,5661	0,5328 – 0,5660	0,9153
1,75	1	0,0333	0,5994	0,5661 – 0,5993	0,5018
1,77	1	0,0333	0,6327	0,5994 – 0,6326	0,4628
1,96	2	0,0666	0,6993	0,6327 – 0,6992	0,1371
2,17	1	0,0333	0,7326	0,6993 – 0,7325	0,1202
1,18	1	0,0333	0,7659	0,7326 – 0,7658	0,8892
2,20	1	0,0333	0,7992	0,7659 – 0,7991	0,8809
2,22	1	0,0333	0,8325	0,7992 – 0,8324	0,1322
2,25	2	0,0666	0,8991	0,8325 – 0,8990	0,2262
2,30	1	0,0333	0,9324	0,8991 – 0,9323	0,5930
2,41	2	0,0666	1	0,9324 – 0,9999	0,8114

* MS Excel uygulamasıyla rassal sayılar elde edilmiştir.

5.9. Örnek (Deneme) Sayısının Belirlenmesi

Örnek sayısının belirlenmesi konusundan araştırmanın daha önceki bölümlerinde bahsedilmiştir.

Örnek sayısı, zaman ve maliyet ile sınırlıysa düşük sayıda tutulabilir; ancak zaman ve maliyet gibi bir sınırlama yoksa örnek sayısının fazla olması istatistiksel testlerde olduğu gibi simülasyon uygulamalarında da daha güvenilir sonuçlar verecektir.

Ortalama yardımıyla örnek sayısı belirlemek istenirse;

$$N = \frac{\sigma^2(Z_{\alpha/2})}{d^2}$$

formülü kullanılabilir.

Burada d gerçek anakütle ortalamasıyla örnek ortalaması arasındaki farktır. Anakütle ile ilgili bilginiz olmadığından d 'nin tahmin edilmesi gerekmektedir. Bunun için de,

$$d = \frac{Z_{\alpha/2} \cdot s}{\sqrt{n}}$$

formülü kullanılır¹²⁰. Bu formülleri kullanarak gerekli örnek sayısını,

$$d = \frac{Z_{\alpha/2} \cdot s}{\sqrt{n}} = \frac{1,65 \cdot \sqrt{0,36}}{\sqrt{30}} = 0,1807$$

$$N = \frac{\sigma^2(Z_{\alpha/2})}{d^2} = \frac{0,36 \cdot 1,65^2}{(0,1807)^2} = 18,1915 \cong 19$$

olarak hesaplarız. Simülasyon uygulaması için 19 gözlem değeri kullanılacaktır. Bu açıklama aynı zamanda 19 deneme yapılacağı anlamına gelir.

Rassal sayıların içinde bulunduğu aralıklar gelişler arası süre ve servis sürelerinin belirlenmesi için kullanılacaktır. Örneğin Tablo 15'in ilk satırındaki rassal sayı olan 0,9296 gelişler arası sürenin 2,30 olduğu anlamına gelir. Çünkü bu değer 2,30'a denk gelen birikimli olasılık değeri içinde yer almaktadır. Benzer şekilde Tablo 16'nın ilk satırındaki rassal sayı olan 0,6606 değeri de servis süresinin 1,96 olduğu birikimli olasılık aralığı içindedir.

¹²⁰ Sariaslan, s :110

5.10. Monte Carlo Simülasyon Metodunun Uygulanması

Bütün rassal sayılar için karşılık geldiği birikimli olasılık değerleri dikkate alınarak uygulanan Monte Carlo metodu aşağıdadır.

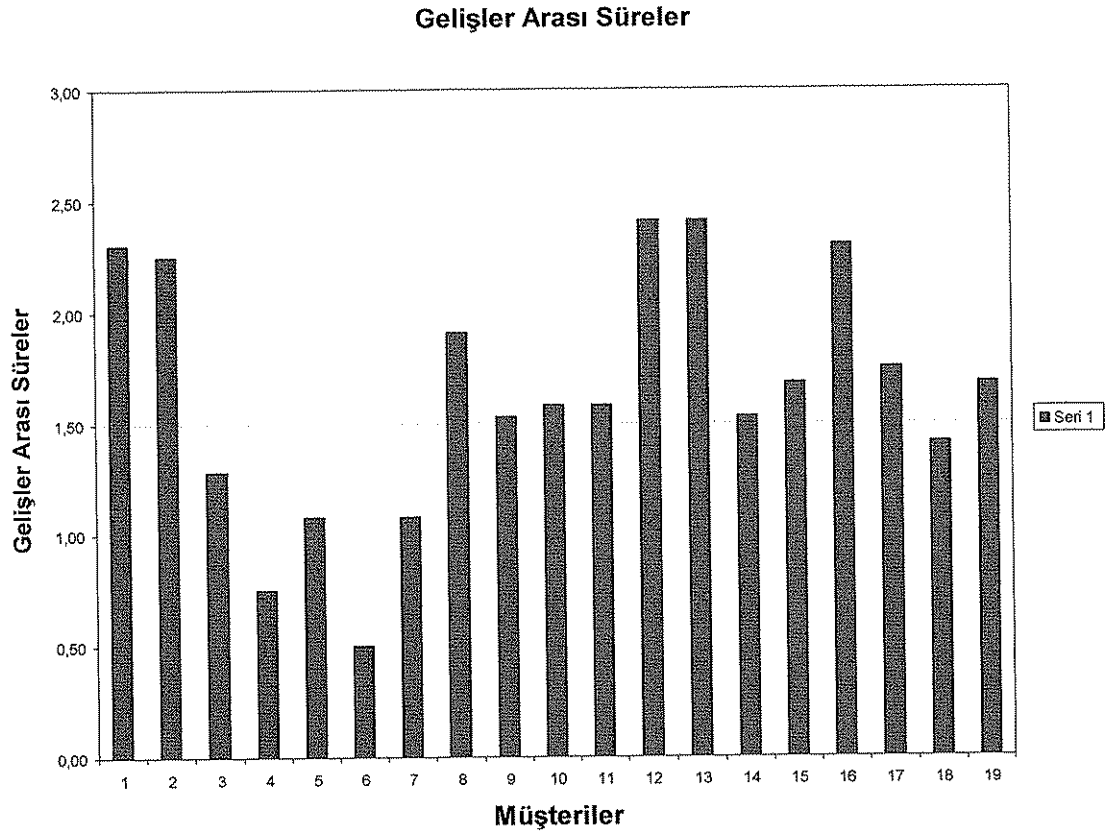
Tablo 17.

Monte Carlo Deneyi Sonucunda Elde Edilen Veriler*

Müşteriler	RS (1)	GAS	Geliş Anı	RS2	Servis Süreleri	S. Başl. A.	S. Bit. A.
1	0,9296	2,30	2,30	0,6606	1,96	2,30	4,26
2	0,8902	2,25	4,55	0,2001	1,09	4,55	5,64
3	0,3164	1,28	5,83	0,4136	1,41	5,83	7,24
4	0,1307	0,75	6,58	0,8136	2,22	7,24	9,46
5	0,1839	1,08	7,66	0,6681	1,96	9,46	11,42
6	0,0398	0,50	8,16	0,4032	1,41	11,42	12,83
7	0,2046	1,08	9,24	0,2299	1,09	12,83	13,92
8	0,8264	1,91	11,15	0,6878	1,96	13,92	15,88
9	0,5677	1,53	12,68	0,8077	2,22	15,88	18,10
10	0,5321	1,58	14,26	0,2371	1,15	18,10	19,25
11	0,5955	1,58	15,84	0,6368	1,96	19,25	21,21
12	0,9563	2,41	18,25	0,4549	1,48	21,21	22,69
13	0,9645	2,41	20,66	0,1722	1,00	22,69	23,69
14	0,4619	1,53	22,19	0,5997	1,77	23,69	25,46
15	0,6532	1,68	23,87	0,2604	1,15	25,46	26,61
16	0,9237	2,30	26,17	0,6349	1,96	26,61	28,57
17	0,7489	1,75	27,92	0,9153	2,30	28,57	30,87
18	0,3739	1,41	29,33	0,5018	1,55	30,87	32,42
19	0,6536	1,68	31,01	0,4628	1,48	32,42	33,90

* Tabloda üretilen rassal sayılar, gelişler arası süreler, servisin başladığı ve bittiği anlar görülmektedir.

Rassal sayıların kullanılmasıyla gelişler arası ve servis süreleri üretilmiştir. Bu süreler araştırmanın başındaki gözlem değerlerinin yerine kullanılarak yukarıdaki tablo oluşturulmuştur .



Şekil 15. Monte Carlo Deneyi Sonucunda Gelişler Arası Süreleri Gösteren Grafik

Deney sonucunda gelişler arası sürelerin gerçek gözlem değerlerinden farklı olduğu görülür. Gerçek verilerle yapılan gözlemde son değerlerdeki frekansların gittikçe azalmasına rağmen simülasyon çalışmasıyla oluşturulan rassal gelişler arası sürelerde bu sürelerin daha düzgün dağıldığı görülür.

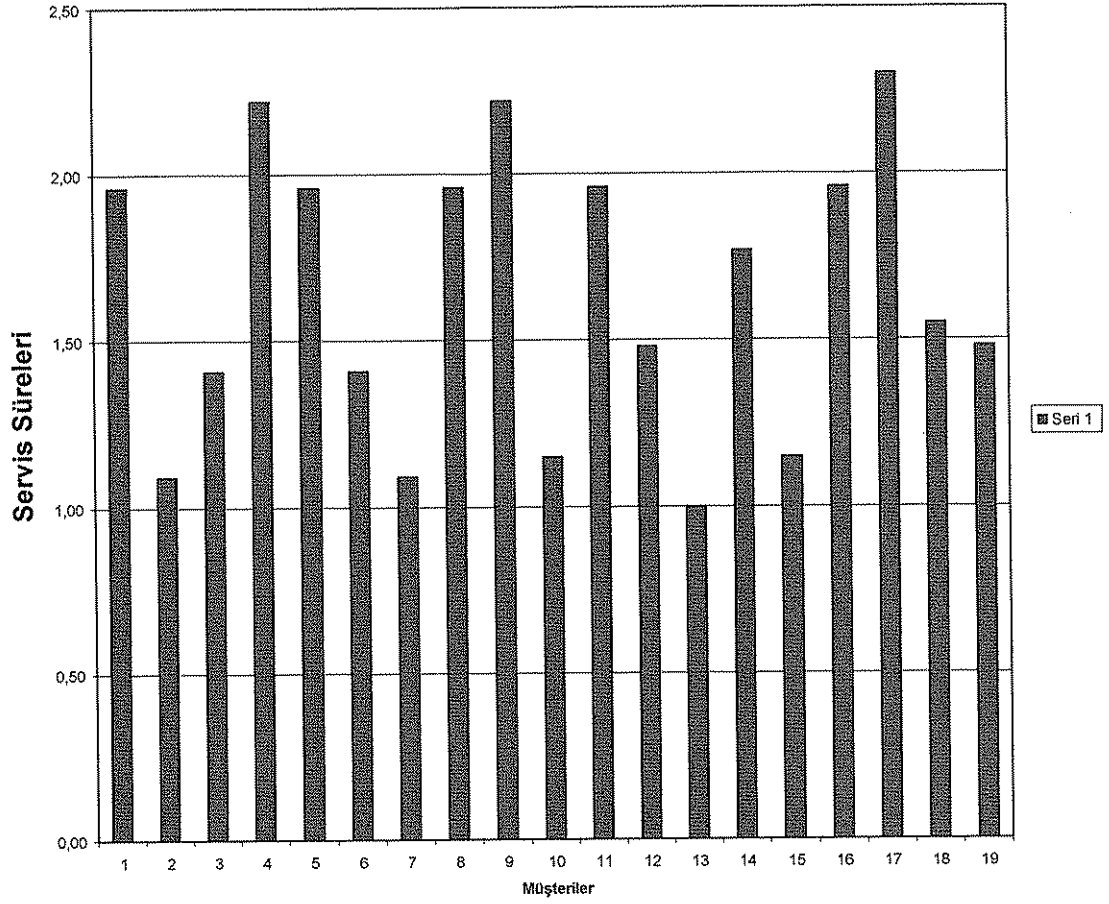
Tablo 18

Monte Carlo Deneyi Sonucunda Elde Edilen Veriler

Müşteriler	SBKS	MSGs	MKGS	KMS	SMS
1	2,30	1,96	0,00	0	1
2	0,29	1,09	0,00	0	1
3	0,19	1,41	0,00	0	1
4	0	2,88	0,66	1	2
5	0	3,76	1,80	3	4
6	0	4,67	3,26	3	4
7	0	4,68	3,59	2	3
8	0	4,73	2,77	3	4
9	0	5,42	3,20	2	3
10	0	4,99	3,84	2	3
11	0	5,37	3,41	2	3
12	0	4,44	2,96	2	3
13	0	3,03	2,03	1	2
14	0	3,27	1,50	1	2
15	0	2,74	1,59	1	2
16	0	2,40	0,44	1	2
17	0	2,95	0,65	1	2
18	0	3,09	1,54	1	2
19	0	2,89	1,41	0	1

Rassal sayılar kullanılarak elde edilen verilerle yukarıdaki tablo oluşturulmuştur. Tabloda sunucunun boş kaldığı süreler, müşterinin sistemde geçirdiği süre, müşterinin kuyrukta geçirdiği süre, kuyruktaki müşteri sayısı ve sistemdeki müşteri sayısı bulunmaktadır.

Servis Sürelerini Gösteren Grafik



Şekil 16. Monte Carlo Deneyi Sonucunda Servis Sürelerini Gösteren Grafik

Araştırmada servis süreleri için gözlem verilerin kullanılması ve bu sürelerin rassal sayılarla oluşturulması arasındaki fark görülmektedir.

Gelişler arası sürelerde olduğu gibi servis sürelerinde de küçük ve büyük değerlerdeki frekansların küçük olmasına rağmen rassal sayılar kullanılarak oluşturulan servis sürelerindeki frekansların daha düzgün dağıldığı görülür.

Tablo 19.**Monte Carlo Deneyi Sonucunda Elde Edilen Veriler**

Müşteriler	λ	μ	W	Wq	L	Lq	Po
1	2,30	1,96	1,96	0,00	1,00	0,00	0,83
2	2,28	1,53	1,53	0,00	1,00	0,00	0,10
3	1,94	1,49	1,49	0,00	1,00	0,00	0,07
4	1,65	1,67	1,84	0,17	1,25	0,25	0,00
5	1,53	1,86	2,22	0,49	1,80	0,80	0,00
6	1,36	1,68	2,63	0,95	2,17	1,17	0,00
7	1,32	1,59	2,92	1,33	2,29	1,29	0,00
8	1,39	1,64	3,15	1,51	2,50	1,50	0,00
9	1,41	1,70	3,40	1,70	2,56	1,56	0,00
10	1,43	1,65	3,56	1,91	2,60	1,60	0,00
11	1,44	1,68	3,72	2,05	2,64	1,64	0,00
12	1,52	1,66	3,78	2,12	2,67	1,67	0,00
13	1,59	1,61	3,73	2,12	2,62	1,62	0,00
14	1,59	1,62	3,69	2,07	2,57	1,57	0,00
15	1,59	1,59	3,63	2,04	2,53	1,53	0,00
16	1,64	1,61	3,55	1,94	2,50	1,50	0,00
17	1,64	1,65	3,52	1,86	2,47	1,47	0,00
18	1,63	1,65	3,49	1,85	2,44	1,44	0,00
19	1,63	1,64	3,46	1,82	2,37	1,37	0,00

Tabloda kuyruk teorisinin ilgilendiği sistem performans ölçüleri görülmektedir. Bu değerler M/M/1 modeli oluşturularak elde edilemezdi. Çünkü hem teorisinin varsayımları sağlanmadığından güvenilir olmayan sonuçlar elde edilirdi hem de gelişler arası sürelerin servis sürelerinden fazla olduğu yinelemelerde bu performans ölçütleri hesaplanamazdı.

Dolayısıyla bir simülasyon modeli kurmak ve bu model üzerinde çalışmak kuyruk teorisine oranla daha güvenilir ve daha gerçekçi bir alternatif üretmiştir.

6. SONUÇLAR

Yapılan çalışmada yöneylem araştırmasının iki önemli konusu üzerinde durulmuş, Kuyruk Teorisi ve Monte Carlo simülasyon yöntemi açıklanmıştır. Yöntemlerin teorik olarak varsayımlarından bahsedilmiş ve bir işletme uygulaması üzerinde çalışma yapılarak araştırmaya son verilmiştir.

Araştırmanın ilk bölümünde kuyruk teorisi ana hatlarıyla açıklanmış, gelişlerarası süreler ve servis süreleri dikkate alınarak karşılaşılabilecek farklı modeller anlatılmış ve sistem performans ölçütlerinden bahsedilmiştir.

Araştırmanın ikinci bölümünde simülasyon yöntemi, üçüncü bölümündeysse Monte Carlo metodunun uygulanışı, rassal sayıların üretilme yöntemleri, test edilmesi konuları üzerinde durulmuştur.

Araştırmanın son bölümünde, küçük çaplı bir işletmede tek kanallı tek servis sağlayıcısı olan bir model üzerinde gözlem yapılarak veriler toplanmış, kuyruk modellerinden M/M/1 modeline uyacağı düşüncesiyle gelişler arası sürelerin Poisson dağılımına, servis sürelerinin de üssel dağılıma uygunluğu araştırılmıştır.

Gelişler arası sürelerin Poisson dağılımına, servis sürelerinin de üssel dağılıma uymaması üzerine kuyruk modellerinden M/M/1 modelinin uygulanamayacağına karar verilmiştir. Uygulama açısından kuyruk modeli oluşturup bu model üzerinde çalışmaya devam etme zorunluluğumuz olsaydı gelişler arası süreler ve servis sürelerinin hangi teorik dağılımdan geldiği test edilip uygun model oluşturulur ve araştırmaya devam edilebilirdi. Gerek bu zorunluluğun bulunmaması gerekse araştırmanın konularından birinin de Monte Carlo simülasyon metodu olmasından dolayı kuyruk modeli yerine bir simülasyon uygulaması üzerinde durulmuştur.

Simülasyon uygulamasına gerekli örnek sayısının belirlenmesiyle başlanmış ve 19 örnekle; diğer bir deyişle 19 deneme ile araştırmaya başlanacağına karar verilmiştir. Daha sonra gelişler arası süreler ve servis süreleri için birikimli olasılık değerleri oluşturulmuştur. MS Office Excel uygulamasının EasyFit eklentisiyle gelişler arası ve servis süreleri için rassal sayılar üretilmiş ve rassal sayıların karşılık geldiği birikimli

olasılık deęerine gre yeniden geliřler arası sreler ve servis sreleri oluřturulmuřtur. Servisin bařladıęı anlar, servis sresi, servisin bittięi an, sunucunun boř kaldıęı sreler, mřterilerin sistemde geirdięi sre, mřterilerin kuyrukta geirdięi sre, kuyruktaki ve sistemdeki mřteri sayıları formllerin girilmesiyle hesaplanmıřtır.

Uygulama aısından asıl nem tařıyan noktaysa kuyruk teorisi ile hesaplanması gereken sistem performans ltlerinin, teorisinin varsayımları saęlanamadıęından dolayı simlasyon uygulamasıyla daha basit bir řekilde hesaplanmasıdır.

Sistem performansını len ortalama geliř hızı (λ), ortalama servis sresi (μ), mřterilerin kuyrukta harcadıkları ortalama sre (W_q), mřterilerin sistemde harcadıęı ortalama sre (W), kuyruktaki ortalama mřteri sayısı (L_q) ve sistemdeki ortalama mřteri sayıları (L) ile servis sunucusunun aylak kaldıęı sreler (P_0) hesaplanmıřtır.

İlk denemelerde geliř hızının 2.30 saat ile bařladıęı, rnek sayısının artmasıyla daha sonraki denemelerde 2.28 saat, 1.94 saat, 1.65 saat, 1.53 saat gibi deęerlerle dengeye gelmeye alıřtıęı ve on dokuzuncu denemede 1.63 saatte sabit kaldıęı grlecektir. Arařtırmaya daha fazla sayıda denemeyle devam edilseydi bu srenin 1.63 veya 1.63'e ok yakın bir sayıyla devam edeceęi grlrd. Zira arařtırmanın ikinci blmnde, simlasyonun ilk ařamalarının gvenilir sonuların elde edilemeyeceęi geici durumdan oluřtuęundan bahsedilmiřti. Ancak rnek sayısının arttırılması ile geici durumdan dzgn duruma geilir.

Sisteme geliř hızında olduęu gibi ortalama servis srelerinde de aynı durum gzlenmiřtir. Ortalama servis sresinin ilk denemede 1.96 saat, ikinci denemede 1.53 saat, nc denemede 1.49 saat olarak hesaplanmıřtır. Deneme sayısı arttırıldıķa, onuncu ve on dokuzuncu denemeler arasında ortalama srenin 1.59 ve 1.68 saat arasında deęerler aldıęı grlmektedir. On dokuzuncu denemede ortalama servis sresi 1.64 saat deęerini almıřtır.

Ortalama servis srelerinin ortalama geliř hızının nne gemesiyle sistemde kuyruk oluřtuęu grlmřtir. Sadece ilk  deneme dıřında btn denemelerde mřterilerin kuyrukta bekledikleri ve bu srenin gittike arttıęı grlmřtir. Drdnc

denemede müşterinin kuyrukta beklediği ortalama süre 0.17 saat iken onuncu denemede 1.91, on ikinci denemede 2.12 saat ve son denemede de 1.82 saat olduğu görülmüştür. Geliş hızı ve servis sürelerinde olduğu gibi müşterilerin kuyrukta harcadıkları ortalama sürelerde de denemelerin ilk aşamalarında değişken değerler aldığı, on yedinci denemede bu sürenin 1.86 saat, on sekizinci denemede 1.85 ve on dokuzuncu denemede de 1.82 değerini aldığı görülmüştür.

Aynı sonuçlar kuyrukta ve sistemde bekleyen ortalama müşteri sayıları için de geçerlidir. İlk denemelerde kuyrukta bekleyen müşteri olmamasından dolayı bu değer 0 ile başlamış, dördüncü denemede 0.25, beşinci denemede 0.49 on yedinci denemede 1.47 ve son denemede de 1.37 olduğu görülmüştür.

Sistemdeki ortalama müşteri sayısında da benzer sonuçlar elde edilmiştir. Uzayan servis süreleri ve uzayan kuyruk, sistemdeki ortalama müşteri sayısını da etkilemiştir. Dördüncü denemeden sonra artmaya başlayan sistemdeki ortalama müşteri sayısı, dördüncü denemede 1.25 müşteri, beşinci denemede 1.80 müşteri, on ikinci denemede 1.67 müşteri ve son denemede de 2.37 müşteri olmuştur.

Uygulamada sadece ilk üç denemede servis sunucusunun belli sürelerle boş kaldığı görülmüştür. İlk denemede 2.30 saat (İlk müşteri gelinceye kadar geçen süre), ikinci denemede 0.29 saat (İlk müşterinin servisten çıkıp ikinci müşterinin servise gelinceye kadar geçen süre), üçüncü denemede ise 0.19 saattir. Dördüncü denemeden itibaren oluşan kuyruk sebebiyle sunucunun aylak zamanının olmadığı görülmüştür.

Kuyrukların oluşması ve uzaması işletme için istenmeyen durumdur. Bu durumda hem servis sağlayıcısının dikkati dağılacak ve performansı düşecek hem de işlerin birikmesine yol açarak müşteri memnuniyetsizliği, dolayısıyla ciro kaybına neden olacaktır.

Bir müddet sonra da teorik olarak biriken işler planlanamayacak ve içinden çıkılamayacak bir hal alacaktır. (İşletme, haftasonları mesai uygulaması yapmakta ve tamir ve imalat için gelen müşteri sayısının çok az olmasından dolayı hafta içinde biriken işleri tamamlayabilmektedir.)

Bu sorunu çözenin yolu ise ek kanal oluşturmak için maliyet hesabı yapmak veya mevcut sistemin performansını geliştirmeye çalışmak olacaktır. Bunun için vardiya sistemi oluşturulabilir.

Tek servisle sisteme devam edip birden fazla işçi sisteme dahil edilebilir. Bu sayede en azından sunuculara boş zaman yaratılabilir ve gün içinde biriken işlerin ilerleyen saatlerde tamamlanması sağlanabilir.

Ek kanal açmak ise bu iş için oldukça zordur. Gözlemin yapıldığı işyerinin küçük çaplı bir işletme olması ve kullanılan CNC tezgahlarının oldukça pahalı olması ek kanal açma fikrine gölge düşürmektedir. Ancak yine de yapılacak bir maliyet analizi ile bu durum işletme sahibi tarafından değerlendirilmelidir.

EKLER

EK 1 Easyfit Uygulaması İle Gelişlerarası Sürenin Poisson Dağılımına Uygunluğunun Testi Sonucu

EasyFit (Evaluation Version) - Poisson Dağılımına Uygunluk - EasyFit - [F16]

File Edit View Analyze Options Tools Window Help

Project Tree: Data Tables (Table1), Results (FX1, FX2, FX3, FX4, FX5, FX6, Stat1)

Graphs Summary Goodness of Fit

Goodness of Fit - Summary

#	Distribution	Kolmogorov Smirnov		Anderson Darling	
		Statistic	Rank	Statistic	Rank
1	Poisson	0,306466433	1	50,8299406	1

Goodness of Fit - Details [hide]

Poisson [#1]

Kolmogorov-Smirnov

Sample Size	30
Statistic	0,306466433
P-Value	0,005344956
Rank	1

α	0,2	0,1	0,05	0,02	0,01
Critical Value	0,19032	0,21756	0,2417	0,27023	0,28987
Reject?	Yes	Yes	Yes	Yes	Yes

Anderson-Darling

Sample Size	30
Statistic	50,8299406
Rank	1

α	0,2	0,1	0,05	0,02	0,01
Critical Value	1,37490912	1,9286236	2,50175596	3,28919661	3,90741594
Reject?	Yes	Yes	Yes	Yes	Yes

EK 2 EasyFit Uygulaması İle Servis Sürelerinin Üssel Dağılıma Uygunluğunun Testi Sonuçları

EasyFit (Evaluation Version) - Üssel Dağılıma Uygunluk - EasyFit - [Fit]

File Edit View Analyze Options Tools Window Help

Project Tree: Data Tables, Table1, Results, Fit

Graphs Summary Goodness of Fit

Fit	Statistic	P-Value	Rank	Statistic	P-Value	Rank
5 Gamma	0,148950955	3	0,773583622	3	0,13686177	3
6 Gamma (3P)	0,141279179	1	0,471916213	1	0,110178907	2

Goodness of Fit - Details [hide]

Exponential [#3]

Kolmogorov-Smirnov

Sample Size	30
Statistic	0,311734826
P-Value	0,004347247
Rank	6

α	0,2	0,1	0,05	0,02	0,01
Critical Value	0,19032	0,21756	0,2417	0,27023	0,28987
Reject?	Yes	Yes	Yes	Yes	Yes

Anderson-Darling

Sample Size	30
Statistic	5,10325222
Rank	6

α	0,2	0,1	0,05	0,02	0,01
Critical Value	1,37480912	1,9286236	2,50175596	3,28919661	3,90741594
Reject?	Yes	Yes	Yes	Yes	Yes

Chi-Squared

Deg. of freedom	2
Statistic	9,74001924
P-Value	0,007673291
Rank	6

α	0,2	0,1	0,05	0,02	0,01
Critical Value	3,21887582	4,60517019	5,99146455	7,82404601	9,21034037
Reject?	Yes	Yes	Yes	Yes	Yes

EK 3 MS Excel İle Oluşturulan Simülasyon Çalışması

[illegible]

KAYNAKÇA

Kitaplar

- Banks, Jerry, John S. Carson II, Barry Nelson ve David M. Nicol. **Discrete Event System Simulation**. 3. Basım. New Jersey – ABD: Prentice Hall International Inc., 2001.
- Binder, Kurt. **Applications Of Monte Carlo Methods To Statistical Physics**. İngiltere: IOP Publishing Ltd., 1997
- Breuer, L. ve D. Baum. **An Introduction To Queueing Theory And Matrix – Analytic Methods**. Hollanda: Springer, 2005.
- Budnick, Frank S., Richard Mojenja ve Thomas E. Vollman. **Principles Of Operations Research For Management**. 18. Basım. Abd: Homewood III, R. D. Irvin, 1977.
- Cinemre, Nalan. **Yöneylem Araştırması**. 2. Basım. İstanbul: Beta Basım Yayım Dağıtım A. Ş., 2004.
- Dagpunar, J. S.. **Simulation And Monte Carlo With Applications In Finance And MCMC**. Chichester – İngiltere: John Wiley&Sons Ltd., 2007.
- Dowdy, Shirlery, Stanley Weardon ve Daniel Chilko. **Statistic For Research**. 3. Basım. New Jersey – ABD: John Wiley&Sons Inc., 2004.
- Everest, Graham ve Thomas Ward. **An Introduction To Number Theory**. ABD: Springer, 2005
- Fan, Xitao, Akos Felsovalyi, Stephan A. Sivo ve Sean C. Kenan. **SAS For Monte Carlo Studies A Guide For Quantitative Researchers**. 1. Basım. Carolina – ABD: SAS Institute Inc., 2002.
- Fine, Benjamin Ve Gerhard Rosenberger. **Number Theory, An Introduction Via The Distribution Of Primes**. ABD: Birkhauser, 2007
- Fishman, George S.. **Monte Carlo Concepts, Algorithms, And Applications**. 3. Basım. ABD: Springer, 1996
- Frederick, S. Hillier ve Gerald J. Lieberman. **Introduction To Operations Research**. 7. Basım. Boston – Abd: The McGraw – Hill Companies Inc. , 2001.

- Gelenbe, Erol ve Guy Pujolle. **Introduction To Queueing Networks**. 2. Basım. Chichester - İngiltere: John Wiley&Sons Inc., 1998.
- Gentle, James E.. **Random Number Generation And Monte Carlo Methods**. 2. Basım. New York – ABD: Springer, 2003.
- Gross, Donald ve Carl M. Harris. **Fundamentals Of Queueing Theory**. 3. Basım. New York – ABD: John Wileysons Inc., 1998.
- Halaç, Osman. **Kantitatif Karar Verme Teknikleri**. 4. Basım. İstanbul: İstanbul Üniversitesi Yayınları, 1983.
- Hendry, David F.. **Handbook Of Econometrics**. 2. Basım. ABD: Elsevier Science Publishers, 1984
- Hsu, Hwei P.. **Theory And Problems Of Probability, Random Variables, And Random Processes**. ABD: Mcgraw Hill, 1997.
- Jackel, Peter. **Monte Carlo Methods In Finance**. İngiltere: John Wileysons Inc., 2002
- Kalos, Malvin H. ve Paula A. Whitlock. **Monte Carlo Methods**. 2. Basım. New York - ABD: Wiley-Interscience, 2004.
- Koralov, Leonid B. ve Yakov G. Sinai. **Theory Of Probability And Random Processes**. 2. Basım. ABD: Springer. 2007.
- Landau, David P. Ve Kurt Binder. **A Guide To Monte Carlo Simulations in Statistical Physics**. 2. Basım. Cambridge – İngiltere: Cambridge University Press, 2000
- Lim, Ghjan ve Joseph Nebus. **Vorticity, Statistical Mechanics, And Monte Carlo Simulation**. ABD: Springer, 2007
- McLeish, Don L.. **Monte Carlo Methods In Finance**. İngiltere: John Wiley&Sons Ltd., 2000.
- Mitrani I.. **Modelling Of Computer And Communication Systems**. 8. Basım. Cambridge – İngiltere: Cambridge University Press, 1987.
- Montgomery, Douglas C. Ve George C. Runger. **Applied Statistics And Probability For Engineers**. 3. Basım. ABD: John Wiley&Sons Inc., 2003.
- Mun, Johnathan. **Modeling Risk – Applying Monte Carlo Simulation, Real Options Analysis, Forecasting, And Optimization Techniques**. New Jersey – ABD: John WileySons Inc., 2005
- Nguyen, Hung T.. **An Introduction To Random Sets**. ABD: Taylor&Francis Group, LLC, 2006.

- Niederreiter, Harald ve Denis Talay. **Monte Carlo And Quasi – Monte Carlo Methods**. Hollanda: Springer, 2006
- Sariaslan, Halil. **Sıra Bekleme Sistemlerinde Simülasyon Tekniği**. Ankara: Ankara Üniversitesi Siyasal Bilgiler Yayınları, 1986.
- Schiff, Maurice. **Introduction To Communication Systems Simulation**. ABD: Artech House Inc., 2006.
- Schriber, Thomas J. **An Introduction To Simulation Using GPSS/H**. New York – ABD: John Wiley&Sons, 1991.
- Shao, Jun. **Matemactical Statistics**. 2. Basım. New York- ABD: Springer, 1998
- Sobol, Ilya M.. **A Primer For The Monte Carlo Method**. 5. Basım. ABD: CRS Press, 1994
- Taha, A. Hamdy. **Yöneylem Araştırması**. 6. Basım. Ş. Alp Baray, Şakir Esnaf (Çev.). İstanbul: Literatür Yayınları, 2000.
- Ulucan, Aydın. **Yöneylem Araştırması, İşletmecilik Uygulamalı Bilgisayar Destekli Modelleme**. 2. Basım. Ankara: Siyasal Kitabevi, 2007.
- Winkler, Gerhard. **Image Analysis, Random Fields And Dynamic Monte Carlo Methods**. 2. Basım. New York – ABD: Springer, 1991
- Yücel, M. Nadir. **Monte Karlo Metodu**. İstanbul: Özarkadaş Matbaası, 1973.

Sürekli Yayınlar

- Adan, Ivo ve Jacques Resing, **Queueing Theory**, Eindhoven University Of Technology, 14 Şubat 2001.
- Coddington, P. D., **Analysis of Random Number Generators Using Monte Carlo Simulation**, Northeast Parallel Architectures Center, 14 Ekim 1993.
- Hançerlioğulları, Aybaba. **Monte Carlo Simülesyon Metodu Ve Mcnp Kod Sistemi**, Kastamonu Eğitim Dergisi. C: 14, Sayı: 2: 548, Ekim 2006.
- Mendeş, Mehmet. **Uygun Simülasyon Sayısının Belirlenmesi: Monte Carlo Simülasyon Çalışması**. Tarım Bilimleri Dergisi. 04 Haziran 2004.
- Metropolis, Nicholas ve S. Ulam, **'The Monte Carlo Method'**, Journal Of The American Statistical Association, Vol:44, No:247, Eylül 1949
- Özdil, Tuncay ve Cengiz Yılmaz, **İstanbul Menkul Kıymetler Borsasında Sektör Bazında İşlem Gören Hisse Senetlerinin Alım – Satım Kararlarında En Yüksek Getirili Stratejinin Belirlenmesi.**, Yönetim Ve Ekonomi. C:13. Sayı 2 : 4, 2006.
- Willig, Andreas. **A Short Introduction To Queueing Theory**. Technical University Berlin, 21 Temmuz 1999.

İnternet Belgeleri

Dinçer, Fehmi. Proje Değerlendirme Sürecinde Çağdaş Bir Yaklaşım: Monte Carlo Simülasyonu, 22 Ocak 2007, [Http://Fehmidincer.Blogspot.Com/2007/01/Nide-İli-Tarihine-Bir-Baki-Fehmi-Diner.Html](http://Fehmidincer.Blogspot.Com/2007/01/Nide-İli-Tarihine-Bir-Baki-Fehmi-Diner.Html), (31 Ocak 2008)

Eckhardt, Roger. Stan Ulam, John Von Neumann, And The Monte Carlo Method, Los Alamos Science Special Issue, 1987, [Http://Library.Lanl.Gov/Cgi-Bin/Getfile?00326867.Pdf](http://Library.Lanl.Gov/Cgi-Bin/Getfile?00326867.Pdf) (31 Ocak 2008)

Simülasyon Dilleri Ve Kullanılan Paket Programlar, 31 Ekim 2005, [Http://Www.Elektrotekno.Com/About20844.Html](http://Www.Elektrotekno.Com/About20844.Html), (21 Ocak 2008)

Simülasyon ve Modelleme Nedir?, 10 Ekim 2006, [Http://Enm.Blogcu.Com/1179385/](http://Enm.Blogcu.Com/1179385/), (23 Ocak 2008)

The RAND Corporation, A Million Random Digits With 100000 Normal Deviates, The RAND Corporation, 1955, [Http://Rand.Org/Pubs/Monograph_Reports/MR1418/Index2.Html](http://Rand.Org/Pubs/Monograph_Reports/MR1418/Index2.Html), (30 Ocak 2008)