SİMÜLASYON

Prof.Dr. Ünal H. ÖZDEN

İSTATİSTİKSEL SİMÜLASYON

- •Simülasyon nedir?
- •Simülasyonun amaçları
- •Simülasyonun avantajları
- •Simülasyonun dezavantajları
- •Uygulama alanları
- •Sistem
- •Sistem çeşitleri

Simülasyon

- "Simülasyon, gerçek sistemin yapısı ve davranışını anlayabilmek için mantıksal ve matematiksel ilişkiler içeren, sistem dışında bilgisayar veya başka araçlarla deney yapma olanağı sağlayan bir yöntemdir."

Simülasyonun Özellikleri

- Sistem davranışlarını gözler ve tanımlar.
- Gözlenen davranışlar için geçerli olan teori ve hipotezleri kurar.
- Sistem davranışlarını öngörür..
- Sistemi kontrol etme olanağı sağlar.
- Simülasyon, karmaşık sistemlerin tasarımı ve analizinde kullanılır

Simülasyon

1)(1978) Case Western Reserve Universitesinde-Yöneylem Araştırması Bölümünde yüksek lisans öğrencileri arasında yapılan bir araştırma sonucunda; **simülasyon** 15 teknik arasında aşağıda görüldüğü gibi **5. sırada** yer almıştır.

1. **İstatiksel metotlar**
2. **Tahmin**
3. **Sitem analizi**
4. **Bilişim sistemeleri**
5. **Simülasyon**

Aynı çalışmanın doktora öğrencileri ile ilgili bölümünde ise ;

"İstatiksel metotlar" birinci sırada olmak üzere **"doğrusal programlama" ile** "**simülasyon" ikinci sırayı** paylaşmaktadır.

Simülasyon ⟹

2)Thomas ve Do Costa (1979),

137 firma arasında yapılan bir ankette **simülasyonin** bu firmaların **%84'ü** tarafından kullanıldığını belirlemiştir. **"İstatiksel Analiz",** ise **%93** kullanım oranı ile 1. sıradadır.

3) Shanon, Long ve Buckles (1980),

A.B.D. De Endüstri Mühendisleri Topluluğunun YA(OR) Bölümündeki üyeleri arasında bir araştırma yapmıştır. Bu araştırma sonuçlarına göre, simülasyon **12** metot arasında, doğrusal programlamadan sonra **2. sırada** yer almıştır.

4)Forgionne (1983) ve Harpell, Lane ve Monsour (1989)

büyük şirketler arasında yaptığı bir araştırmada, **sekiz farklı metot** arasında **simülasyonin 2.sırada** yer aldığını göstermiştir.

Bu araştırmaların tümü, o yıllarda simülasyonin kullanımının hızla yaygınlaştığını göstermektedir. Bu gelişmeye en büyük katkıyı, simülasyon yazılımları ve bilgisayar teknolojisindeki hızlı gelişmeler sağlamaktadır.

Simülasyonun Amaçları

Simülasyon aşağıda verilen amaçlardan birisini veya bir kaçını gerçekleştirmek için kullanılır.

- **Değerlendirme :** Belirlenen kriterlere göre önerilen sistemin ne kadar iyi çalıştığının gösterilmesi
- **Karşılaştırma :** Önerilen sistem tasarımlarının veya politikaların karşılaştırılması
- **Tahmin :** Önerilen koşullar altında sistemin performansının tahmin edilmesi
- **Duyarlılık Analizi :** Sistemin performansı üzerinde hangi faktörlerin etkili olduğunu belirlenmesi
- **Optimizasyon :** En iyi performans değerini veren faktör düzeylerinin bir kombinasyonunun belirlenmesi
- **Darboğaz Analizi :** Bir sistemde darboğazların belirlenmesi amacıyla (Pedgen et all, 1995)

simülasyon kullanılır.

Ne Zaman Kullanılır?

Bazı problemlerin çözümünde simülasyonun kullanılması zorunluluğu ortaya çıkabilir. Aşağıdaki durumlarda simülasyon kullanılmasında yarar vardır (Shannon, 1975, s.11):

- Problemin tam matematiksel modelinin olmaması.
- Matematiksel modelin analitik yaklaşımla çözülememesi.
- Analitik çözümün mümkün fakat bu çözümün matematiksel olarak çok karmaşık olması.
- Analitik çözümün maliyetleri artırması.
- Sistem henüz tasarım aşamasında ise
- Sistemin davranış analizi yapılacaksa

Simülasyon kullanılır

İyi Bir Simülasyon Çalışması Nasıl Olmalıdır?

İyi bir simülasyon çalışması için aşağıdaki koşulların sağlanması gerekir (Shannon, 1975, s. 22):

- Kullanıcılar tarafından kolay anlaşılmalıdır.
- Amaçlar veya hedefler doğru tanımlanmalıdır.
- Güçlü ve güvenilir olmalı, model hatalı sonuçlar vermemelidir.
- Kolayca dönüştürülebilir ve kontrol edilebilir olmalıdır.
- Kolayca değişikliklere uyarlanabilmeli kontrol edilebilmeli ve güncelleştirilebilmelidir.

Dikkat!

- Hiç bir model tam olarak gerçek sistemin aynısı olamaz. Dolayısıyla simülasyon modeli kurulurken unutulan veya göz ardı edilen bazı ayrıntılar veya yapılan bazı varsayımlar simülasyon sonuçlarını etkileyebilir.
- Simülasyon çalışmasında insan faktörünün olması simülasyon sonuçlarını etkileyebilir.
- Tüm analiz yöntemlerinde olduğu gibi simülasyon çalışmasında da hatalı verilerin kullanılması, parametrelerin yanlış belirlenmesi ve soyutlamanın yanlış yapılması hatalı sonuçlara neden olur.

Simülasyonun Dezavantajları

- Simülasyon, analitik yöntemlere göre daha maliyetlidir ve uzun zaman alır.
- Analitik yöntemlere göre daha fazla uzmanlık isteyen bir yöntemdir.
- Simülasyon, incelenen sistemin etkinliği hakkında sadece sayısal bilgiler verebilir. Sebep-sonuç ilişkileri hakkında sayısal verilerden görülebilecek ipuçları dışında bilgi vermez.
- Uygun olmayan varsayımlar, modeli gerçek sistemden uzaklaştırır ve yanlış kararlar alınmasına neden olur.

Simülasyonun Uygulama Alanları

Genel olarak uygulama alanları :

1. **BİLGİSAYAR SİSTEMLERİ:** Yazılım sistemleri, bilgisayar ağları, veri tabanı yapısı ve yönetimi, bilgi işleme, donanım ve yazılım güvenliğinde
2. **ÜRETİM :** Malzeme taşıma sistemleri, montaj hatları, otomatik üretim tesisleri, otomatik depolama tesisleri, stok kontrol sistemleri, fabrika yerleşimi, makina tasarımı
3. **İŞLETME :** Stok ve mal analizi, ücretlendirme politikası, pazarlama stratejileri, nakit akış analizleri, tahmin, ulaştırma alternatifleri, işgücü planlaması
4. **KAMU HİZMETİ:** Askeri silahlar ve kullanımları, askeri taktikler, nüfus tahmini, arsa kullanımı, sağlık hizmetleri, polis servisleri, itfaiye hizmetleri, karayolu tasarımı, trafik kontrolü
5. **EKOLOJİ VE ÇEVRE :** Su kirliliği ve temizlenmesi, atık kontrolü, hava kirliliği, hava tahmini, deprem ve fırtına analizi, maden arama ve çıkarma, güneş enerjisi sistemleri, tahıl üretimi.
6. **SOSYOLOJİ :** Yiyecek/ nüfus analizi, eğitim politikaları, organizasyon yapısı, sosyal sistemlerin analizi, refah sistemleri, üniversite eğitimi
7. **BİYOLOJİ :** Salgın hastalık kontrolü, biyolojik yaşam çevrimi, biomedikal çalışmalar. (Pegden, Shanon & Sodowski, 1995)

Simülasyon Paket Prog. ve Simülasyon Dilleri

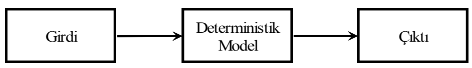
Özellik	Simülasyon Paket Prog.	Simülasyon Dili
Esneklik	Sınırlı	Sınırsız
Geliştirme Çabukluğu	Hızlı	Yavaş
Uygulama Alanı	Belirli uygulamalar	Sınırsız
Sistem Detayları	Önceden Tanımlanmış	Sınırsız
Modelin Değiştirilebilirliği	Sınırlı	Sınırsız
Tecrübe Gereksinimi	Az	Çok
Programlama gerektirmez	Evet	Hayır
Grafik model oluşturmak için arayüze sahiptir	Evet	Hayır
Tüm yöneticiler ve çalışanlar kolayca kullanabilir	Evet	Hayır
Bazı temel modeller hazırdır	Evet	Hayır
Sistemin tanıtılması için esnektir	Hayır	Evet
Fazla basitleştirmeye neden olabilir	Evet	Hayır

Simülasyon Türleri

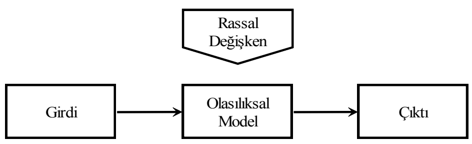
- **Sistem ya da Sürecin Durumuna Göre: Statik ve Dinamik Simülasyon**
- Sistemin zamanın herhangi bir durak noktasındaki durumunu gösteren simülasyon modeline statik simülasyon modeli, bu modelle yapılan simülasyona da statik simülasyon adı verilir. Statik simülasyon modeli ile genellikle Monte Carlo simülasyonu kastedilmektedir. Monte Carlo simülasyonu ileride açıklanacaktır. Montaj hattı dengeleme ve işletmelerin fiziksel konumunu düzenleme statik simülasyon modelleri için örnek gösterilebilir.
- Sistemin zaman boyutundaki gelişmesini gösteren simülasyon modeline dinamik simülasyon modeli, bu modelle yapılan simülasyona da dinamik simülasyon denir. Bu modellerdeki değişkenler veya varlıklar zaman içersinde değişim ve etkileşimler gösterirler. Sipariş sistemleri, kuyruk sistemleri, stok sistemleri dinamik simülasyon modelleriyle ifade edilebilir.

Deterministik ve Olasılıksal Simülasyon

- **Simülasyona Esas Olan Değişkenlerin Olasılıksal Olup Olmamasına Göre: Deterministik ve Olasılıksal Simülasyon**
- Simülasyon modeli deterministik veya olasılıksal olabilir. Rassal değişken içermeyen modellerle yapılan simülasyon deterministiktir. Deterministik simülasyon modellerinde hiçbir rassal özellik yoktur. Bu nedenle, simülasyon sonuçları her denemede aynı sayısal değeri verir. Deterministik simülasyon Şekil 2.1'de verildiği gibi işler.
- Olasılıksal simülasyon modelinde bir ya da daha fazla sayıda rassal değişken bulunur. Rassal değişkenler üretilerek çalıştırılan olasılıksal simülasyon modellerinde deney sonuçları da rassal olur. Ölçülmek istenen performans göstergeleri için ise tahmini değerler elde edilir. Olasılıksal simülasyonun işleyişi Şekil 2.2'de gösterilmiştir.



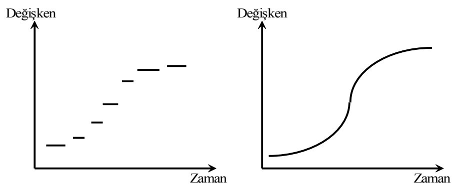
Şekil 2.1: Deterministik Simülasyon



Şekil 2.2: Olasılıksal Simülasyon

Kesikli ve Sürekli Simülasyon

- **Sistem Değişkenlerinin Değişiminin Zaman İçinde Gözlenmesine Göre: Kesikli ve Sürekli Simülasyon**
- Sürekli simülasyon söz konusu olduğunda, sistemin durumunu belirleyen değişkenlerin değerleri zaman içersinde sürekli değişir. Bir sıvının bir borudan akması veya nüfusun değişimi gibi sistemler sürekli simülasyonun ilgi alanına girer.
- Kesikli simülasyonda ise, dikkate alınan değişkenlerin değerleri zamanın belirli noktalarında değişmektedir. Kesikli sistemlerde sistem durumu sadece olaylar kesin olarak meydana geldiğinde değişebilir. Bir kuyruğun uzunluğu veya müşterinin bekleme zamanı istatistikleri sadece yeni bir müşterinin kuyruk sistemine ulaşması ya da sistemi terketmesiyle değişir.

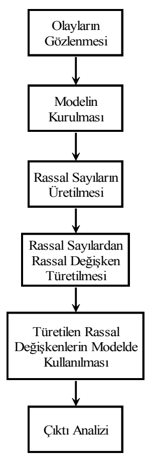


Şekil 2.1: Sürekli ve Kesikli Simülasyon

Monte Carlo Simülasyonu

Monte Carlo Simülasyonu

Monte Carlo adı rulet ve şans oyunlarını çağrıştırmaktadır. Bu isim ilk kez John Von Neumann tarafinan atom bombası geliştirme uğraşları sırasında rassal sayılar üzerine yaptığı çalışmalarda kullanılmıştır (Watson ve Blackstone, 1989, s.7). Monte Carlo simülasyonu günümüzde zaman akışının önemli olmadığı deterministik veya olasılıksal problemlerin çözümünde rassal sayılar kullanılarak yapılan plan olarak tanımlanır. Dinamik karekterli olmaktan çok statik karakterli (Law ve Kelton, 1991, s.113) olan Monte Carlo simülasyonu, en iyi bilinen ve yaygın olarak kullanılan simülasyon yöntemlerinden biri olarak belirginleşmekte ve parametrelerin olasılık dağılımlarıyla modellenebileceği varsayımına dayanmaktadır.



Şekil 2.1: Monte Carlo Simülasyonu Akış diyagramı

Sistem ve Çevresi

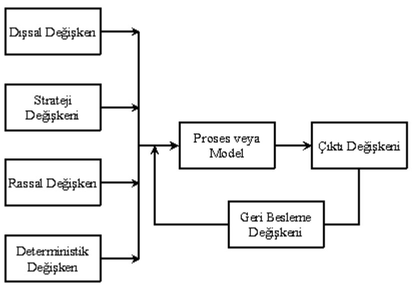
Sistem; bir amacı gerçekleştirmek için aralarında düzenli bir etkileşimin veya bağımlılığın bulunduğu nesneler topluluğudur. Örneğin; Otomobil üreten bir üretim sisteminde, makinalar, iş parçaları ve işçiler; yüksek kalitede bir araç üretmek için birlikte bir **montaj hattı** oluştururlar.

Bir sistem, kendisi dışında ortaya çıkan değişikliklerden etkilenir. Sistemlerin modellerinin kurulabilmesi için, sistem ve sistemin çevresi arasındaki sınıra karar vermek gerekir. Bu karar, sistemin özelliğine ve çalışmanın amacına bağlıdır.

Sistem değişkenleri; sistemin varlıklarının, niteliklerinin, özelliklerinin belirlenmesinde etkide bulunan, değerleri sistemin durumuna göre farklı değerler alabilen özelliklere değişken denir.

Konu sistemin tanımlanması veya simülasyon çalışması olduğunda değişkenler genellikle **çıktı ve girdi değişkenleri** olmak üzere iki grupta incelenir. Dışsal, strateji, rassal ve deterministik değişkenler girdi değişkenleridir. Bu değişkenlere, sistem denetimini sağlayan ve gerçekte bir çıktı değişkeni olan geri besleme değişkeni de eklenebilir. Girdi değişkeni rassal ise çıktı değişkeni de rassaldır. gruplamada yer almamasına karşın konu simülasyon olduğunda değişkenlerin sürekli veya kesikli olarak gruplandırılması önemlidir. Sistem girdilerinin belirli bazı işlemlere tabi tutulması sonucunda farklı bir yapıya dönüşen sistem sonuçlarına ilişkin değişkenlere çıktı değişkenleri denir.

Değişkenler



Şekil 1.1: Sistem Analizi ve Simülasyondaki Değişkenler

Tablo 1.1: Sistem Analizi ve Simülasyondaki Değişkenlerin Değişik İsimleri

Değişken Tipi	Değişik İsimleri
Çıktı	Bağımlı, Tepki, Endojen
Geri Besleme	Seri korelasyonlu, Bağımsız, Girdi
Dışsal	Exojen, Çevresel, Kontrol edilemeyen, Bağımsız, Girdi
Strateji	Karar, Kontrol edilebilir, Bağımsız, Girdi
Rassal	Olasılıksal, Stokastik, Bağımsız, Girdi
Deterministik	Bağımsız, Girdi

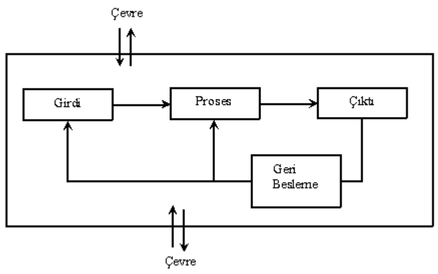
Sistem Bileşenleri

- Değişkenlerin değerleri ve/veya kaynakların nasıl tahsis edileceği üzerine konulan sınırlandırmalara **kısıt** denir. Kısıtlar, sistem analisti veya sistem yöneticisi tarafından konulabileceği gibi sistemin doğası sonucu ortaya çıkabilir. Sistemin doğası sonucu beliren kısıtlar hiçbir şekilde değiştirilemezken karar vericinin koyduğu kısıtlar değiştirilebilir.
- Sistemi oluşturan elemanlar arasındaki her türlü akış **ilişki** olarak adlandırılır. İlişkiler mekansal, zamansal, nedensel, mantıksal, matematiksel, yapısal, işlemsel veya sırasal olabilir.
- Değişkenler arasındaki ilişkileri belirleyen sabit katsayılara **parametre** denir. Parametreler aynı sistemin farklı durumları için farklı değerler alabilir ve matematiksel ilişkilerden yararlanarak hesaplanır.
- Zaman boyutunun herhangi bir t anında varlıkların, özelliklerin ve faaliyetlerin tanımları sistemin **durum**u olarak adlandırılır.
- Sistem belirli bir çevre içerisinde işler ve içinde bulunduğu çevrede oluşan değişikliklerden fazlasıyla etkilenir (Watson ve Blackston, 1989, s.3). **Sistemin çevresi** sistemin durumunu etkileyebilen sistem dışı öğeler ile bunlara ait özelliklerinin kümesidir. Sistemi etkilemeyen dışsal elemanlar sistemin çevresine dahil değildir.

- Sistemin bir durumdan başka bir duruma geçmesine neden olan harekete, işe, etkiye **aktivite** adı verilir. Aktiviteler sistemde değişmelere neden olur. Sistem olayı, belirli uzunluktaki belirli bir zaman süresinde sistemin (veya çevresinin) bir veya daha fazla yapısal özelliklerindeki değişim olarak tanımlanabilir.
- Bir sistemin kendi içindeki ve dıştan gelen etkilere karşı verdiği yanıta sistemin davranışı denir.
- Birbirine bağlı ardışık kararların alınmasında sistemin yerine getirdiği işleri kontrol ederek karar vericiye sistemin durumu hakkında bilgi aktarma sürecine **geri besleme** denir. Genel olarak tüm sistemler geri besleme içerir. Denetim, geri beslemenin amacı olup sistemin standart durumları ile gerçekleşen durumların karşılaştırılmasını ve gözlenmesini gerektirir. Bunlar arasında bir fark olduğunda, daha efektif sonuçlar elde etmek için bilgiler sisteme tekrar alınır.

- ***Girdi*:** Sistem tarafından işlenerek çıktıya dönüştürülen sistem bileşenlerine girdi denir. Dönüştürmede sistemin amaçları dikkate alınır. Girdiler genel olarak üç başlık altında incelenir. Bunlar;
- i. Değişim sürecinden geçen **materyaller**: Bu başlık altındaki girdilere sistem yükü de denir. Ürünün son halini alması için birleştirilen maddeler bu tür girdilerdir.
- ii. Sistemin işlevlerini etkileyen **çevre**: Çevre, sistem tasarımcısının hiç bir şekilde etkileme şansının olmadığı etkilerdir. Yerçekimi bu tür bir girdidir. Bu durumda çevresel girdiler, insanları, fiziksel kaynakları, iklimi, ekonomik ve pazar koşullarını, davranışları ve yasaları kapsamaktadır.
- iii. **Değişim sürecinin kendisini oluşturan girdiler**: Enerji, insangücü ve bilgi gibi girdileri kapsamaktadır.
- ***Proses*:** Bir sistemin bir işlemi yerine getirmesi için başlangıçtan sonuç alıncaya kadar geçen süreçte izlenen uygulama işlevlerine proses denir (Özdamar, 1988, s.31). Proses, sistemi oluşturan davranışlar dizisidir ve hedef üretme işlevi bulunan proses, girdilerin çıktılara dönüştürülmesi işidir. Hem etkin hem de verimli olmak için sistem girdilerine uygulanacak dönüştürmenin en iyi şekilde tasarlanması gerekir.

Girdi, çıktı, proses, geri besleme ve çevre arasındaki ilişkiler Şekil 1.2'de özetlenmiştir.



Şekil 1.2: Basit Sistem İlişkileri

- ***Ölçüt*:** Sistemin hedef ve amaçları ile bunların nasıl değerlendireceği ve yargılanacağının standardına ölçüt denir. Ölçütün yanlış tanımlanması yanlış sonuçlara yol açacağından bu konuya dikkat edilmesi uygun olur.

Sistemlerin Sınıflandırılması

- ***Kesikli-Sürekli Sistem***: Sistem değişkenlerinin zaman içersinde aldıkları değerlerin kesikli veya sürekli olması incelenen sistemin türünün belirlenmesinde önemli bir etkendir.ü
- *Kapalı-Açık Sistem*: Sistemler çevreleri ile olan ilişkilerine göre açık veya kapalı olarak ikiye ayrılır.
- *Statik-Dinamik Sistem*: Gerçek dünyada herhangi bir oluşum genel olarak zaman-dan bağımsız düşünülemez. Zaman içinde sistem durumunun değişip değişmemesine göre sistemler statik ve dinamik diye ikiye ayrılır.
- *Canlı-Cansız Sistem*: Biyolojik özelliklere sahip sistemlere canlı sistemler denir. Bu tür sistemler doğum, ölüm ve çoğalma gibi özelliklere sahiptirler. Cansız sistemler ise biyolojik yapıya sahip olmayan sistemlerdir.
- *Somut-Soyut Sistem*: Bileşenlerinin yapısına göre sistemler somut ve soyut sistemler olmak üzere iki başlık altında incelenir.
  - i. Somut Sistem: Sistem bileşenlerinden en az ikisi somut varlık olan sisteme somut sistem denir.
  - ii. Soyut Sistem: Tüm bileşenleri kavram olan sistemlere soyut sistemler denir.

- *Durum Koruyucu Sistem*: Durum koruyucu sistemler aynı içsel veya dışsal bir olay gerçekleştiğinde aynı, farklı bir olay gerçekleştiğinde farklı bir tepki gösterirler. Bu tür sistemler sistem durumunu dengede tutarlar. Durum koruyucu sistemlere termostat iyi bir örnektir.
- *Amaç Arayışlı Sistem*: Durum koruyucu sistemin aksine, sistemi belirli bir durumda sabit tutmak için değil sistemi halihazırda var olmayan bir duruma ulaştırmak için işlev gören sisteme amaç arayışlı sistem denir.
- *Birleşik Sistem*: Birleşik sistemler birlikte işleyen iki veya daha fazla alt sistemden oluşur. Diğer bir ifade ile, birleşik sistem birden fazla alt sistemin her birine ait prosesin tüm sistemin amacını gerçekleştirmek için ayrık olarak birlikte çalıştığı sistemdir. Alt sistemler birbirlerine seri, paralel veya karışık biçimde bağlanabilirler. Birbirinden bağımsız olarak çalışan birden fazla alt sistemin söz konusu olduğu ve bunların çıktılarının bir diğer sistemin girdilerini oluşturması durumundaki sistemlere paralel sistem denir.
- *Doğal ve Yapay Sistem*: Sistemler oluşum ya da meydana geliş biçimlerine göre doğal ve yapay sistemler olmak üzere iki grupta toplanır. Doğal ve yapay sistemler somut ya da soyut olabilir.

Sistem Yaklaşımı

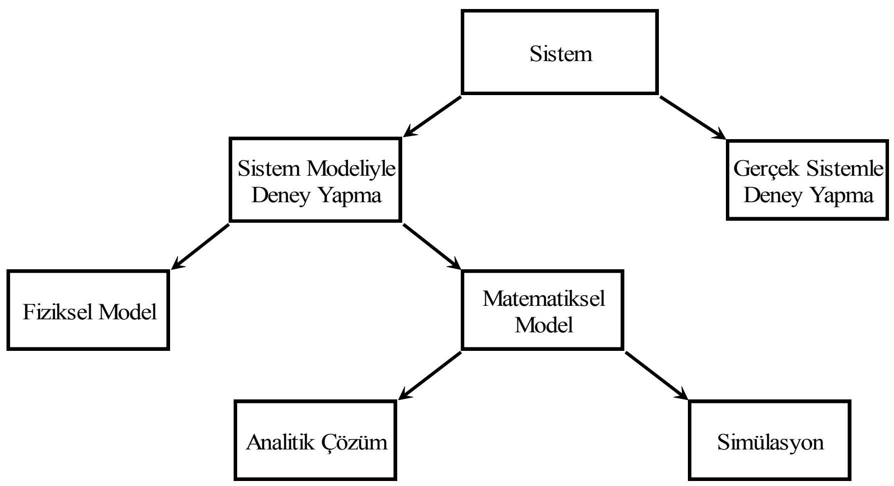
Sistem yaklaşımı, sistem ve alt sistemlerin en verimli biçimde bütünleştirilmesini kapsar. Bunun için mühendislik, ekonomi, davranış bilimleri, matematik, istatistik, yöneylem araştırması vb. bilimleri biraraya getiren, sorunu bir bütün olarak ele alıp, amaçları düzenleyen ve sonucu genelleyici bir yaklaşımla değerlendiren disiplinlerarası bir yaklaşımdır. Sistem yöneticilerine kavramsal bir çerçeve çizmek amacı güden yaklaşım, en yalın biçimiyle belirli hedeflere ulaşmak için olası yolların incelenmesi olarak tanımlanabilir. İlke olarak sistemi süreçlerin bir bütünü olarak inceleyen sistem yaklaşımında, analitik yaklaşımdaki öğelerin yerini süreçler, değerlerin yerini davranışlar alır.

- Sistem yaklaşımı bütünsel bir yaklaşımdır.
- Sistem yaklaşımı akıl yürütmeye ve matematiğe dayalı disiplinlerarası bir yaklaşımdır.
- Sistem yaklaşımı bilimsel bir yaklaşımdır. Karar verme ve sorun çözme bilimsel bir yöntemle gerçekleştirilir.
- Sistem yaklaşımı türetilebilirlik özelliğine sahiptir. Belirli bir sorunun çözümünde kullanılan yöntemler, benzer problemlerin çözümünde bazı değişiklikler yapılarak kolayca kullanılabilir.
- Sistem yaklaşımı karar verme sürecinde sezgi, deneyim ve nicel sonuçlardan yararlanır.
- Sistem yaklaşımı sistemdeki her bileşeni ve sorunu evrimsel olarak ele alır.

Analitik ve sistem Yaklaşımının Karşılaştırılması

Analitik Yaklaşım	Sistem Yaklaşımı
Sistem elemanlarını önce ayrıştırır daha sonra toplar.	Sistem elemanları arasındaki etkileşimlere bir bütün olarak yaklaşır ve onlar üzerine yoğunlaşır.
Etkileşimlerin doğasını araştırır.	Etkileşimlerin sonuçlarını araştırır.
Detayları anlama üzerinde durur.	Bütünsel anlama üzerinde durur.
Belirli bir zamanda yalnızca bir değişkenin değerinin değiştiği kabul edilir.	Gerçek hayatta olduğu gibi tek bir değişken değil değişken gruplarının hepsi bir arada değişebilir.
Zamandan bağımsızdır, dikkate alınan olaylar tersine çevrilebilir.	Zamanı ve tersine çevrilemeyen olayları dikkate alır.
Deneysel olarak kanıtlanmış teorinin yapısı içinde, sistem durumlarının geçerliliği araştırılır.	Model davranışı ile gerçekliği karşılaştıra-rak durumların geçerliliği araştırılır.
Gerçek ilişkiler için daha az kullanışlı olan kesin ve ayrıntılı modelleri (ekonometrik modeller gibi) kullanır.	Karar vermede ve kararın uygulanmasında oldukça faydalıdır.
Etkileşim doğrusal ve zayıf olduğunda etkin-dir.	Etkileşim güçlü ve doğrusal değilse etkin-dir.
Disiplin odaklı eğitime öncülük eder.	Disiplinlerarası eğitime öncülük eder.
Ayrıntıda programlanmış faaliyetleri yönlendirir.	Faaliyetlerden amaçlara kadar olan herşe-ye öncülük eder.
Zayıf belirlenmiş hedefler söz konusu oldu-ğunda ayrıntı bilgisine sahiptir.	Ayrıntılar bulanık olduğunda kullanılır.

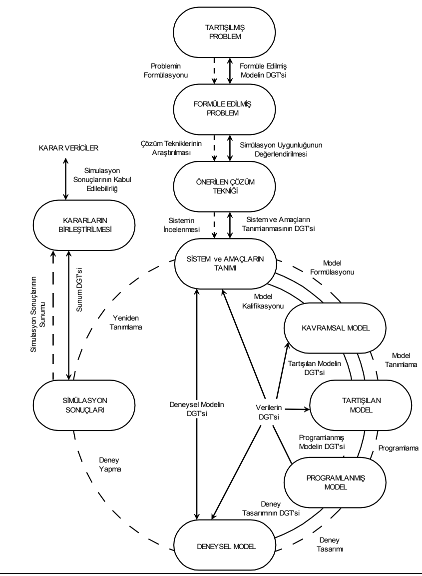
Sistem Analizinde Kullanılan Bazı Yaklaşımlar



Simülasyon Modelleme Süreci

- Birçok alanda kullanılan simülasyon disiplinlerarası bir yaklaşımdır ve model tasarımı, model çalıştırma ve model analizinde kullanılır. Simülasyon amacıyla kurulan modele simülasyon modeli denir. Simülasyon modeli genellikle sistemin işleyişi hakkındaki varsayımlar doğrultusunda sistemin önemli bileşenleri arasındaki ilişkileri matematiksel veya mantıksal sembollerle açıklayan bir kümedir
- Simülasyon "yaparak öğrenme" ilkesine dayandığından, sistemi öğrenmek için önce sistemin modellenmesi sonra kurulan modelin çalıştırılması gerekir. Sistemin gerçekliğini ve karmaşık yapısını anlamak için oluşturulan yapay varlıkların dinamik faaliyetlerdeki rolleri incelenir.
- Farklı amaçlar için problem çözme ve deney yapmaya yönelik yürütülen simülasyon çalışmaları, problemin formüle edilmesi ve simülasyon çalışmasının sonuçlarının tanıtılması ile sona erer. Formülasyon süreci analiz, modelleme ve deney yapmayı içerir. Başarılı bir simülasyon çalışması yöneylem araştırması, bilgisayar, istatistik ve mühendislik gibi farklı disiplinler hakkında asgari düzeyde de olsa bilgi sahibi olmayı gerektirir.

Simülasyon Modellem Süreci (Hayat Döngüsü)



Doğrulama, Geçerlilik ve Test etme

- Simülasyon çalışmasının hayat döngüsü ve hayat döngüsü boyunca simülasyonun doğruluğunun belirlenmesinde kullanılan prensiplerle ilgili açıklamalar aşağıda verilmiştir. Simulasyon çalışmasının kalitesi doğrulama (verification), geçerlilik (validation) ve test etme (testing) ile belirlenir.
- *Model Doğrulama*: Modelin doğru kurulmasıyla ilgili olan model doğrulama. belirli bir formdan başka bir forma dönüştürülen modelin yeterli bir doğruluk ve dikkatle gerçekleştirilmesi işlemidir. Formüle edilen problemin belirli bir modele dönüştürülmesi veya tanımlanan modelin bilgisayar programının doğruluğuna ilişkin yapılan çalışmalar doğrulamanının kapsamına girer. Bir modelin doğruluğunu kanıtlamak modelin temsil ettiği sistemin tam ve doğru bir kopyası olduğunu garanti etmek anlamına gelir. Ayrıca doğrulama faaliyeti, modelin sistemle ilgili belirlemelere uygun olarak kurulmasını, yapısındaki ve algoritmasındaki hataların giderilmesini kapsar.
- *Model Geçerliliği*: Modelin geçerliliği, modelin kullanım alanı içinde çalışmanın amaçları ile yeterli doğrulukta tutarlı olduğunu kanıtlamak için gerçekleştirilir. Geçerliliğin sınanması ile gerçek sistem ile onun temsilcisi olan model arasındaki uyum araştırılır. Doğruluğun değerlendirilmesinde model bilgisayarda veya zihinsel olarak işletilerek ortaya çıkan model davranışları sistem davranışları ile karşılaştırılır. Bunlar uyum içinde ise modelin geçerliliği sağlanmış demektir. Model varsayımlarının ve sonuçlarının geçmişe yönelik, geleceğe yönelik, yapısal ve yöntemsel olarak geçerliliği araştırılmalıdır.

Doğrulama, Geçerlilik ve Test etme

- *Model Testi*: Modelin hatalı veya yanlış olup olmadığı modelin test edilmesiyle araştırılır. Modelin testi verilerin veya olayların kendi fonksiyonlarını yerine getirip getirmediğinin belirlenmesine yöneliktir. Testler doğruluğun, geçerliliğin ya da her ikisinin araştırılmasında kullanılır. Bazı testler davranışsal doğruluğu (geçerlilik), bazıları da modellerin bir formdan başka forma dönüştürülmesinin doğruluğunu (doğrulama) araştırır. Modelin geçerliliğinin ve doğruluğunun test edilmesi sürecine model doğrulama, geçerlilik test (DGT) süreci denir.
- *Karar* (Balcı, 1998a): Belirli bir amaç için geliştirilen simülasyon modelinin kullanılabilir ve güvenilir olduğunun organizasyon tarafından kabul edilmesi olarak tanımlanır.

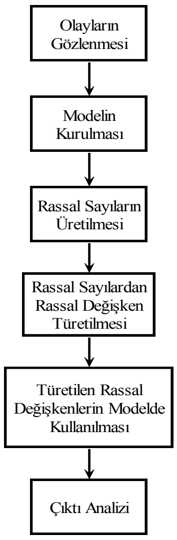
Rassal Sayılar

Değeri 1 ile N arasında bulunan ve seçilme olasılığı 1/N olan sayıya rassal sayı denir. Buna göre rassal sayı belirli koşulları gerçekleyen stokastik bir değişkendir. Bir değişkenin rassal değişken olabilmesi için gerçeklemesi gereken iki koşul vardır. Bunlar:

1. Sıfırla bir aralığında tekdüze (düzgün) dağılırlar. Sayıların gerçekleşme olasılıkları birbirlerine eşittir.
2. Bu sayılardan oluşan dizi istatistiksel bağımsızlık göstermeli, yani her bir rassal sayının gerçekleşme olasılığı geçmişte ortaya çıkan sayıların gerçekleşme olasılıklarından bağımsız olmalıdır.

Bu koşulları sağlayan stokastik değişkenler U ile gösterilecektir. Bir başka ifade ile rassal sayılar ui ile ifade edilecektir.

Simülasyon Süreci



Şekil 2.1: Monte Carlo Simülasyonu Akış diyagramı

Rassal Sayı Üretme

İlk zamanlarda rassal sayıların üretimi para, zar, iskambil kağıdı, rulet gibi fiziksel araçlar kullanılarak gerçekleştirilmekteydi. Günümüzde de zaman zaman bu araçlara başvurulduğu görülmektedir. Bu araçlardan hangisi kullanılırsa kullanılsın genelde bu yolla çok miktarda sayı üretimi gerçekleştirilememekte, ayrıca işlem çok zaman almaktadır.

Geniş ölçekli simülasyon çalışmalarında çok sayıda rassal sayıya ihtiyaç duyulduğundan 1940'lı yıllarda araştırmacılar rassal sayı üretmek için matematiksel yaklaşımlara yönelmişlerdir. İlk matematiksel üreteç von Neumann ve Metropolis tarafından ortaya atılan orta kare yöntemidir. Günümüzde bilgisayardan da yararlanarak matematiksel yaklaşımlarla rassal sayılar daha hızlı ve istenilen miktarda kolayca üretilebilmektedir.

Literatürde fiziksel araçlarla üretilen rassal sayılara gerçek rassal sayılar, matematiksel yaklaşımlarla üretilenlere ise sahte rassal sayılar denir.

Rassal Sayı Üreteçlerin Taşıması Gereken Özellikler

Rassal sayı üretmek için kullanılan fiziksel araçların veya yöntemlerin sahip olması gereken özellikler kısaca aşağıdaki başlıklarla açıklanabilir.

- Rassallık tek tek sayıların değil sayılar dizisinin özelliğidir.
- Sayıların tekrarlanma peryodu uzun olmalıdır. Bir başka deyişle daha önce üretilen bir rassal sayının tekrarlanması olabildiğince geç olmalıdır.
- Üretilen rassal sayıların dizisi minimum varyanslı olmalıdır.
- Üretilen rassal sayılar birbirlerinden ardışık olarak bağımsız olmalıdır. Yani u_i, kendisini izleyen sayı (u_{i+1}) hakkında hiçbir bilgi vermemelidir.
- Önceden üretilen sayılar istendiğinde yeniden üretilebilmelidir.
- Bilgisayarda bir algoritma kullanılarak üretim yapılacaksa, üretilen sayılar bilgisayarın gelişmişliğine bağlı olmamalıdır.
- Üretilen rassal sayılar herhangi bir X değişkeninin asimtotik dağılımına kolayca uyabilmelidir.
- Üretim kısa sürede gerçekleşmelidir.

Fiziksel Araçlarla Rassal Sayı Üretme

Fiziksel araçlar az sayıda rassal sayı üretilmek istendiğinde kullanılabilir. Ancak bu araçlarla istenildiği zaman istenildiği kadar rassal sayı üretmek mümkün olmadığı gibi aynı çalışmayla ilgili veya benzer türden bir simülasyon için aynı rassal sayıları tekrar üretmek mümkün değildir (Law ve Kelton, 1991, s.421). Yeniden üretilebilme simülasyon sonuçlarının varyansının küçültülmesi için önemlidir (Kleijnen ve Groenandaal, 1992, s.17). Rassal sayı üretiminde kullanılan fiziksel araçlardan bazıları aşağıdaki gibidir.

a) Yazı Tura Atma
b) Zar Atma
c) Rulet
d) Tablolar

Fiziksel Araçlarla Rassal Sayı Üretme

a) Yazı Tura Atma

Bir madeni paranın yüzlerinden birine 1, diğerine 0 değeri verilir. Para atılır ve gelen yüze bağlı olarak ya 1 veya 0 üretilir. Bu işlemin n kez tekrarlanması sonucunda sadece 0 ve 1 değerlerinden oluşan n elemanlı bir dizi oluşturulur. Bu sayıların her biri her seferinde (2^n – 1)'in ikili sistemde karşılık gelen sayısına bölünerek 2^n adet rassal sayı belirlenmiş olur. Bölme işlemiyle elde edilen sayıların değeri 0 ile 1 arasında değişir.

b) Zar Atma

Rassal sayı üretmede kullanılan araçlardan biri de yüzleri 0, 1, ..., 9 olarak numaralandırılmış 10 yüzü bulunan zardır. Tıpkı para atışında olduğu gibi zar atılır ve üste gelen yüzdeki sayı not edilir. Zarın ilk atılışında belirlenen sayı ilk ondalık basamakta bulunan sayıyı verir. Kaç tane ondalık basamak isteniyorsa zar atışı o sayıda tekrarlanır ve her tekrarda belirlenen sayı arka arkaya yazılır.

c) Rulet

Rulet ile rassal sayı üretmek için yuvarlak olan rulet alanı 100 eşit parçaya ayrılır. Her bir parça 00'dan 99'a kadar bir tam sayı ile numaralandırılır. Sayıların gerçekleşme olasılıkları 0,01'dir. Rulet ile iki basamaklı rassal sayı üretme işlemi çok basit olmasına rağmen sayıların üretilme hızı son derece yavaştır

Fiziksel Araçlarla Rassal Sayı Üretme

d) Tablolardan Yararlanarak Rassal Sayı Üretme Yöntemleri

Rassal sayı tabloları rassal sayı üretmek amacıyla kullanılan araçların başında gelir. Değişik zamanlarda farklı kişi ve kurumlarca yayınlanmış rassal sayı tabloları vardır (Law ve Kelton, 1991, s.421). Bunlar arasında en önemlilerinden biri yaygın olarak kullanılan Rand Corporation tarafından 1955 yılında yayınlanan bir milyon rassal sayının yer aldığı tablodur. Bu tablo birçok uygulamacı ve araştırmacı tarafından rassal sayılar tablolarının en iyilerinden biri olarak kabul edilmektedir.

Rassal sayı tablolarının kullanımı şöyledir: Rassal sayı çekmek için herhangi bir başlangıç noktası rassal olarak belirlenir. Daha sonra satırda ilerleyerek veya sütundan aşağıya doğru inerek istenen miktarda sayı çekilir. Örneğin ondalık kısmı on rakamdan oluşacak rassal sayı için bu tablodan 10 adet sayı alınır. Bir sonraki rassal sayı için bir sonraki 10 sayı alınır. Belirlenen sayılar rassal sayıların virgülden sonraki kısmına sırayla yazılır (Kleijnen ve Groenandaal, 1992, s.18).

Bu tablolar daha çok elle gerçekleştirilen simülasyonlarda kullanılmaktadır. İleride açıklanacağı gibi simülasyon sonuçlarından maksimum fayda sağlayabilmek için simülasyonun çok fazla tekrarlanması gerekir. Bu çok fazla sayıda rassal sayı üretilmesi gerektiği anlamına gelir. Bu yüzden simülasyonun elle değil bilgisayar kullanımıyla gerçekleştirilmesi zorunludur. Bilgisayarla simülasyonda kullanılan teknikler aşağıda açıklanmıştır.

Sahte Rassal Sayı Üretme

Bilgisayarla rassal sayı üretimi belirli bir algoritma veya yinelemeli matematiksel ilişkinin kullanımıyla gerçekleştirilir. Yinelemeli ilişkinin kullanılması rassal sayı u_i'nin kendisinden önce üretilen rassal sayılarla (u_{i-1}, u_{i-2}, ...) ilişkili olmasına yol açar. Matematiksel ilişki deterministik olduğundan bu yolla üretilen rassal sayılara sahte rassal sayılar denir. Bununla birlikte matematiksel ilişki tam olarak bilinmiyorsa ve/veya iyi belirlenmemişse bir sonraki rassal sayı bulunamaz (Kleijnen ve Groenendaal, 1992, s.18). Sahte rassal sayı üretiminde kullanılabilecek çok sayıda algoritma vardır.

Orta Kare Yöntemi

Orta kare yönteminde (Von Neuman, 1940) ilk olarak kaç haneli sayı üretilecekse o kadar haneli Z_0 gibi bir başlangıç sayısı rassal olarak seçilir. Daha sonra Z_0'ın karesi alınarak en fazla 2m haneli bir sayı elde edilir. Kare alma işlemiyle belirlenen sayı 2m haneli değilse bu sayının sol tarafına hane sayısını 2m yapacak kadar sıfır eklenir. Hane sayısı 2m olan bu sayının sağından ve solundan m/2 kadar hane atılarak, ortada kalan sayı bir yere yazılır. Yazılan sayının (0,1) aralığında bulunan sayıya dönüştürülmesi için sayı 10^m'e bölünür. Bölme işlemiyle elde edilen sayı orta kare yöntemiyle üretilen ilk rassal sayı (u_1) olur. İkinci rassal sayı için u_1'in ondalık kısmı Z_0 gibi kabul edilerek aynı işlemler tekrarlanır ve ikinci rassal sayı üretilir (Kleijnen ve Groenendaal, 1992, s.19). Kaç tane sahte rassal sayı isteniyorsa yöntem o sayıda tekrarlanır. İstenilen miktarda sayıya ulaşıldığında uygulama tamamlanmış olur.

Bu yöntemi kullanmanın bazı sakıncaları aşağıdaki gibi sıralanabilir:

- Üretilen sayıların tekrarlanma periyodu çok kısadır.
- Başlangıç sayısı uygun seçilmemişse üretim kısa sürede sıfıra ulaşmaktadır.
- Çok miktarda sayı üretildiğinde sayıların frekans dağılımında ritmik yığılmalar olmaktadır.
- Başlangıç sayısı kişinin seçimine bağlı olduğundan tam olarak rassal değildir.

Orta Kare Yöntemi-Örnek

Kaç haneli sayı üretilecekse (hane sayısı m), o kadar haneli Z_0 gibi bir başlangıç sayısı rassal olarak seçilir.

Z_0 = 3387 başlangıç sayısı olsun

$(Z_0)^2$= 10156969

(Z_1)= 1569

$(Z_1)^2$= 02461761

(Z_2) = 4617

$(Z_2)^2$= ?

Doğrusal Eşlik Yöntemi

Bu yöntemde (Lehmer, 1951) rassal olarak belirlenen bir tamsayı Z_0 başlangıç sayısı (tohum) olarak alınır.

Doğrusal eşlik yöntemi ile rassal sayı üretmiminde kullanılan ilişki aşağıda gösterilmiştir.

$Z_i = (aZ_{i-1} + b) \bmod m$ (i = 0, 1, ..., m-1) (4.5)

Burada m modüler aritmetik böleni, a çarpan ve b de eklenecek sabit sayıdır. m, a, b ve Z_0 negatif olmayan tamsayılar olup 0 < m, a < m, b < m ve Z_0 < m koşullarını gerçeklemeleri gerekir (Law ve Kelton, 1991, s.25). Bu yöntemin algoritması aşağıda açıklandığı gibidir.

- **Adım 1**: Z_0, a, b ve m değerlerinin belirlenerek denklem (4.5)'te yerlerine konulması.
- **Adım 2**: a'nın Z_{i-1} ile çarpımına b'nin eklenmesi sonucu bulunan değerin m'ye bölünmesi.
- Bölme sonucunda kalan sayı Z_i'ye eşittir. Böylece rassal sayı üretiminde kullanılacak ilk değer Z_1 elde edilmiş olur. Z_i bölme sonucunda kalan sayı olduğundan $0 \leq Z_i \leq (m - 1)$ ilişkisi söz konusudur (Law ve Kelton, 1991, s.425).
- **Adım 3**: Bir sonraki Z değeri için, ikinci adımda elde edilen Z_i'in formülde yerine konulması ve ikinci adımdaki matematiksel işlemlerin yapılması.
- Bu adım üretilmek istenen rassal sayı miktarı kadar tekrarlanır.
- **Adım 4**: Üretilen Z_i değerlerinin üretim sırasına göre sıralanması ve her birinin ayrı ayrı m'ye bölünmesi ($u_i = Z_i/m$).

Doğrusal Eşlik Yöntemi-Örnek

$Z_i = (aZ_{i-1} + b) \bmod m$ (i = 0, 1, ..., m-1)

Burada

m: modüler aritmetik böleni,

a: Çarpan

b: Eklenecek sabit sayı

m, a, b ve Z_0 negatif olmayan tamsayılar olup $0 < m$, $a < m$, $b < m$ ve $Z_0 < m$, koşullarını gerçeklemeleri gerekir.

$Z_o = 0$, a = 2, b=3, m=10 olarak belirlenmiş olsun.

	$u_i = Z_i/m$
$Z_1 = (2x0 + 3) \bmod 10 = 3$	$u_1 = Z_1/m=3/10=0{,}3$
$Z_2 = (2x3 + 3) \bmod 10 = 9$	$u_2 = Z_2/m=9/10=0{,}9$
$Z_3 = (2x9 + 3) \bmod 10 = 1$	$u_3 = Z_3/m=1/10=0{,}1$
$Z_4 = (2x1 + 3) \bmod 10 = 5$	$u_4 = Z_4/m=5/10=0{,}5$
……	……….

Eklemeli Eşlik Yöntemi

Eklemeli eşlik yöntemi doğrusal eşlik yönteminden hareketle geliştirilmiş bir yöntemdir. Bu yöntemde doğrusal eşlik yönteminden farklı olarak önceden belirlenmesi gereken tohum sayısı bir tane değil k tanedir. Bunun yanı sıra daha önce üretilmiş olan rassal sayı miktarını çoğaltılmak istendiğinde de eklemeli eşlik yönteminden yararlanılabilir. Yöntem aşağıdaki matematiksel ilişkinin kullanılmasıyla uygulanır.

$Z_i = (Z_{i-1} + Z_{i-k}) \bmod m$ (i = 0, 1, ...) (4.6)

Burada m (negatif olmayan) modüler aritmetik bölenidir. Eklemeli eşlik yöntemi algoritması aşağıda verilmiştir.

- **Adım 1**: Başlangıç değeri Z_0, m ve k değerlerinin belirlenerek denklem (4.6)'da yerlerine konulması.
- **Adım 2**: Z_{i-1} ile Z_{i-k}'nın toplamı sonucu bulunan değerin m'ye bölünmesi.
- **Adım 3**: Bir sonraki Z değeri için, ikinci adımda elde edilen Z_i'nin formülde yerine konulması ve ikinci adımdaki matematiksel işlemlerin yapılması.
- Bu adım üretilmek istenen rassal sayı miktarı kadar tekrarlanır.
- **Adım 4**: Üretilen Z_i değerlerinin üretim sırasına göre sıralanması ve her birinin ayrı ayrı m'ye bölünmesi ($u_i = Z_i/m$). Bu şekilde (0, 1) aralığında tanımlanan rassal sayılar (u_i) üretilmiş olur.

Eklemeli eşlik yöntemiyle üretilen sayılar genelde birbirleriyle ilişkili çıkar. Bu nedenle uygulamada pek fazla kullanılmaz.

Eklemeli Eşlik Yöntemi-Örnek

$Z_i = (Z_{i-1} + Z_{i-k})$ mod m

k=3 ise k tane Z_i değeri başlangıç olarak belirlenmelidir. Bu nedenle,
m=10, Z_1=3, Z_2=7, Z_3=9 olarak belirlenmiş olsun.

	$u_i = Z_i/m$
$Z_4 = (Z_3 + Z_1)$ mod 10 = (9+3) mod 10 = 2	$u_4 = Z_4/m$ =2/10=0,2
$Z_5 = (Z_4 + Z_2)$ mod 10 = (2+7) mod 10 = 9	$u_5 = Z_5/m$ =9/10=0,9
…………	…………….

Karesel Eşlik Yöntemi

Coveyou tarafından geliştirilen karesel eşlik yönteminde de başlangıç olarak m (m ≥ 2) ve Z_0 değeri saptanır. Yöntem aşağıdaki matematiksel ifade kullanılarak uygulanır:

- $Z_i = (Z_{i-1} (Z_{i-1} + 1))$ mod m (i = 0, 1, ...) (4.7)

Bu yöntem başlangıç temel sayı Z_0'ın keyfi olarak seçilmesinden dolayı tam karesel bir yöntem değildir. Ayrıca diğer eşlik yöntemlerinde olduğu gibi bu yöntemle de uzun tekrarlama periyotlu rassal sayılar üretilebilmesi için m ve Z_0'ın seçimi önemlidir.
Karesel eşlik yöntemi algoritması aşağıda verilmiştir.

- **Adım 1**: Başlangıç değeri Z_0 ile m saptanarak ve formül (4.7)'de yerlerine konulması.
- **Adım 2**: Z_{i-1} ile $(Z_{i-1} +1)$ ile çarpımı sonucu bulunan değerin m'ye bölünmesi.
- Böylece rassal sayı üretiminde kullanılacak ilk değer Z_i elde edilmiş olur.
- **Adım 3**: Bir sonraki Z değeri için ikinci adımda elde edilen Z_i'in formülde yerine konulması ve ikinci adımdaki matematiksel işlemlerin yapılması.
- Bu adım üretilmek istenen rassal sayı miktarı kadar tekrarlanır.
- **Adım 4**: Üretilen Z_i değerlerinin üretim sırasına göre sıralanması ve her birinin ayrı ayrı m'ye bölünmesi ($u_i = Z_i/m$). Bölme işlemi sonucu belirlenen sayılar (0, 1) aralığında yer alan sahte rassal sayılardır.

Karesel Eşlik Yöntemi-Örnek

$Z_i = (Z_{i-1} (Z_{i-1} + 1)) \bmod m$

$m \geq 2$ ve Z_0 değeri saptanır

Z_0=2 ve m =16 olarak saptandığını düşünelim

$u_i = Z_i/m$

$Z_1 = (Z_0 (Z_0 + 1)) \bmod 16 = 2x(2+1) \bmod 16 = 6$ $u_1 = Z_1/m = 6/16=0,375$

$Z_2 = (Z_1 (Z_1 + 1)) \bmod 16 = 6x(6+1) \bmod 16 = 10$ $u_2 = Z_2/m = 10/16=0,625$

$Z_3 = (Z_2 (Z_2 + 1)) \bmod 16 = 10x(10+1) \bmod 16 = 14$ $u_1 = Z_3/m = 14/16=0,875$

.... ………………………………..

Çarpımsal Eşlik Yöntemi

Bu yöntem doğrusal eşlik yönteminin özel bir durumudur (Kleijnen ve Groenendaal, 1992, s20). Doğrusal eşlik yönteminin eklenen sabiti b sıfır olduğunda, yöntem çarpımsal eşlik yöntemine dönüşür. Bu yöntemde de öncelikle Z_0 başlangıç değeri, a çarpanı ve m belirlenir. a, Z_0 ve m'nin belirlenmesinde dikkat edilmesi gereken hususlar ve yapılan tanımlamalar doğrusal eşlik yönteminde yapılan açıklamalar ile aynıdır. Çarpımsal eşlik yöntemi ile rassal sayılar aşağıdaki formülden hareketle üretilir:

- $Z_i = (aZ_{i-1}) \bmod m$ (i = 0, 1, ...) (4.8)

Bu yöntemin algoritması aşağıda açıklandığı gibidir.

- **Adım 1**: Z_0, a ve m değerlerinin belirlenerek formülde yerlerine konulması.
- **Adım 2**: a'nın Z_{i-1} ile çarpımı sonucu bulunan değerin m'ye bölünmesi.
- Bölme sonucunda kalan sayı Z_i'e eşittir. Böylece rassal sayı üretiminde kullanılacak ilk değer Z_1 elde edilmiş olur.
- **Adım 3**: Bir sonraki Z değeri için, ikinci adımda elde edilen Z_i'in formülde yerine konulması ve ikinci adımdaki matematiksel işlemlerin yapılması.
- Bu adım üretilmek istenen rassal sayı miktarı kadar tekrarlanır.
- **Adım 4**: Üretilen Z_i değerlerinin üretim sırasına göre sıralanması ve her birinin ayrı ayrı m'ye bölünmesi ($u_i = Z_i/m$). Bölme işlemi sonucu belirlenen sayılar (0,1) aralığında yer alan sahte rassal sayılardır.

Çarpımsal Eşlik Yöntemi-Örnek

$Z_i = (aZ_{i-1}) \bmod m$

Başlangıç olarak a, Z_0 ve m değerlerinin hesaplanması gerekir.

a = 3, $Z_0 = 5$ ve m =10 olsun,

	$u_i = Z_i/m$
$Z_1 = (aZ_0) \bmod 10=3x5 \bmod 10 =5$	$u_1 = Z_1/m =5/10=0,5$
$Z_2 = (aZ_1) \bmod 10=3x5 \bmod 10 =5$	$u_2 = Z_2/m =5/10=0,5$
………………….	………………

Tausworthe Yöntemi

Şifreleme metodlarıyla ilgili olan bu yöntem Tausworthe'un 1965 yılında yazdığı makaleden hareketle yaratılmıştır. Bu yöntemde rassal sayılar b_1, b_2, ... gibi ard arda gelen sayı çiftlerinin tekrarlanmasıyla üretilir.

- $b_i = (c_1b_{i-1} + c_2b_{i-2} + ... + c_qb_{i-q}) \bmod 2$ (4.9)

Burada c_1, c_2, ..., c_q'ler, 0 veya 1 olan sabitlerdir. Bunlardan en fazla 2 tanesi sıfırdan farklı olabillir. Bu nedenle yukarıdaki bağıntı en basit haliyle,

- $b_i = (b_{i-r} + b_{i-q}) \bmod 2$ (4.10)

olarak yazılır. Burada r ve q tamsayı ve $0 < r < q$ olmalıdır.
Bu bağıntı yardımıyla, 0–1 sayılar (ikili sistem sayıları) dizisi oluşturulur. Bu diziden rassal sayı üretmek için dizi, aynı uzunlukta (k) komşu alt dizilere bölünür. Alt dizinin uzunluğu (k) < q olmalıdır. Alt dizileri oluşturan ikili sistem sayıları 10'lu sisteme dönüştürülerek alt dizi sayısı kadar tamsayı yaratılır. Bu tamsayıların 2^k'ya bölünmesi ile uygulama tamamlanmış olur.
Bu yöntemin doğrusal eşlik yöntemlerine göre bazı avantajları vardır. Bu yöntemde üretilen sayıların kullanılan bilgisayardan bağımsız ve çok uzun periyotlu sayılar üretmek mümkündür (Law ve Kelton, 1991, s.435). Ayrıca Jhonson (1987, ss.41-42) bu yöntemle çok değişkenli rassal vektörler üretilebileceğini belirtmektedir.

Tausworthe Yöntemi-Örnek

Fibonacci Yöntemi

Fibonacci yöntemi eklemeli eşlik yönteminin özel bir durumudur. Eklemeli eşlik daha belirgin olan bir yöntemdiryönteminin matematiksel ifadesinde bulunan k indisi 1 olduğunda, yöntem Fibonacci yöntemine dönüşür (Kleijnen ve Groenendaal, 1992, s.21). Ancak ona oranla teorik temelleri . Fibonacci yöntemi, Fibonacci serisine dayanır ve

- $Z_i = (Z_{i-1} + Z_{i-2}) \bmod m$ (4.11)

olarak gösterilir (Morgan, 1984, ss.51-56).

Bu yöntemin algoritması ile eklemeli eşlik yönteminin algoritması aynıdır. Bu yöntemle genellikle m'den daha büyük süre peryodlu sayı üretilir. Ancak Fibonacci yöntemi ile üretilen rassal sayılar rassallık testlerini geçmeyi başaramazlar. Sonuç olarak bu yöntem tatmin edici sonuçlar vermediği için pek kullanılmaz.

Fibonacci Yöntemi-Örnek

$Z_i = (Z_{i-1} + Z_{i-2})$ mod m

Shuffling Yöntemi

Birçok araştırmacı birbirinden farklı iki veya daha fazla rassal sayı üretme yöntemlerini birleştirerek yeni yöntemler türetmişlerdir. Bunlardan en çok bilineni shuffling yöntemidir. Yöntem 1965 yılında MacLaren ve Marsaglia tarafından ortaya atılmış, Marsaglia ve Bray (1968) tarafından geliştirilmiştir. Bu yöntem iki farklı yöntemin birleştirilmesine dayanmaktadır. Birinci yöntemle üretilen rassal sayı Z_1 (m'e bölünmemiş) ile ikinci yöntemle üretilen rassal sayının Z_2 (m'e bölünmemiş hali) farkı alınıp, m (tamsayı) değerine bölünerek yeni rassal sayılar üretilir. Matematiksel olarak

- $Z_i = (Z_{1i} - Z_{2i})$ mod m (4.12)

şeklinde gösterilir. Rassal sayılar için Z_i'ler m'ye bölünür. Bu yöntemle üretilen sayıların tekrarlama periyodu oldukça uzundur (L'Ecuyer, 1988). Bununla birlikte çok yavaş çalışması ve uygulanmasının zor olması yöntemin dezavantajıdır (Kleijnen ve Groenendaal, 1992, s. 22).

Shuffling Yöntemi-Örnek

$Z_i = (Z_{1i} - Z_{2i}) \mod m$

Rassallık Testleri

Simülasyon çalışmalarında önemli olan sahte rassal sayıların nasıl üretildiğinden çok, bu sayıların gerçekten rassal olup olmadıklarıdır . Sahte rassal sayıların gerçek rassal sayılarda doğal olarak bulunan rassallık özelliğine, sahip olup olmadığı, rassallık testleriyle kontrol edilir. Sahte rassal sayılar deterministik olarak üretilse de, simülasyonda kullanılabilmeleri için rassal olmaları gerekir. Simülasyon diliyle konuşulurken telaffuz edilen rassal sayı, değeri 0 ile 1 arasında bulunan ve üretilme şansı bu aralıktaki diğer bütün sayıların üretilme şansına eşit olan sayıdır. Bu özellik kısaca U(0,1) olarak gösterilir. Bu nedenle sayıların rassallığını incelemeden önce sayıların tekdüze dağılıma uygun biçimde dağılıp dağılmadıkları araştırılmalıdır. Sayıların U(0, 1) olması onların rassallığınının garantörü değildir. Ardışık olarak üretilen sayılar arasında ilişki bulunmaması gerekir. Özetle, sahte rassal sayıların gerçek rassal sayılarla aynı istatistiksel özellikler taşıması beklenir. Bu amaçla geliştirilmiş testlere tabi tutulmaksızın sahte rassal sayıların kullanılması doğru olmaz.
Rassallık testlerinin en çok kullanılanları şunlardır:

- Koşu Testleri
  - Aşağı ve Yukarı Doğru Koşu Testi
  - Ortalamaya Göre Koşu Testi
  - Koşu Uzunluğu Testi
- Korelasyon Testi
- Poker Testi
- Gap Testi

Koşu Testleri

Koşu testlerine geçmeden önce konuyla ilgili bazı kavramlar üzerinde durulması uygun olur. Birbirini izleyen ve öncesi ile sonrasında farklı olaylar gerçekleşen aynı olayların oluşturduğu diziye koşu, koşu içindeki olayların sayısına da koşu uzunluğu denir. Koşu testlerinde ya koşuların sayısı ya da koşuların uzunluğu dikkate alınır.

Koşu testlerinde öncelikle sahte rassal sayıların üretilme zamanlarına göre sıralanması gerekir. Bu işlemin tamamlanmasının ardından belirli kriterlere göre yeni bir dizi oluşturulur ve rassallık analizi bu yeni diziye uygulanır. Test yeni dizi değerlerinin bağımsızlığına işaret ederse, üretilen rassal sayıların bağımsız olduğuna karar verilir. Farklı koşu testleri vardır. Bunlardan en çok kullanılan üç tanesi aşağıda açıklanmıştır.

1. Yukarı ve Aşağı Doğru Koşu Testi
2. Ortalamaya Göre Koşu Testi
3. Koşu Uzunluğu Testi

Yukarı ve Aşağı Doğru Koşu Testi

Koşuların sayısının dikkate alındığı testtir. Üretilmiş olan n adet rassal sayı kronolojik olarak sıralanır. Sıraya konulmuş her bir sayı kendinden bir önce gelen sayı ile büyüklük ve küçüklük yönünden karşılaştırılır. (i + 1)'inci rassal sayı (i)'inci rassal sayıdan büyük ise yukarı doğru, küçük ise aşağı doğru koşu söz konusu olur ve yeni dizinin değerleri yukarı doğru koşu için 1, aşağı doğru koşu için 0 olarak belirlenir (Watson ve Blackston, 1989, s.158). Kısaca kronolojik olarak sıralanan sahte rassal sayılar aşağıdaki denetlemeden geçirilir.

$$s_i = \begin{cases} 1 & u_i \leq u_{i+1} \text{ ise} \\ 0 & u_i \geq u_{i+1} \text{ ise} \end{cases} \qquad (i = 1, 2, ..., n\text{-}1) \qquad (4.13)$$

Yeni dizide birbirini izleyen k adet s_i değerinin her biri 1'e eşit ise k uzunlukta yukarı doğru koşu, 0'a eşit ise k uzunlukta aşağı doğru koşu söz konusu olur. Koşu sayısının üretilmiş sahte rassal sayıların miktarına eşit olması ile koşu sayısının 1'e eşit olması uç durumlardır. Beklenen koşu sayısı genellikle bu uç durumlar arasında yer alır.

Yukarı ve Aşağı Doğru Koşu Testi-devam

Sahte rassal sayıların sayısı n ise koşu sayısı en fazla (n – 1), en az 1 olur (Banks vd., 1996, s. 304). Koşu uzunluklarının kısa olması sahte rassal sayıların rassal, yani bağımsız olduğuna işaret eder. n çok büyük olduğunda (Knuth'a göre n ≥ 4000) aşağı ve yukarı doğru koşuların toplam sayısının (a) beklenen değeri ile varyansı aşağıdaki gibi hesaplanır.

$$E[a] = \mu_a = \frac{2n-1}{3} \qquad (4.14)$$

$$\sigma_a^2 = \frac{16n-29}{90} \qquad (4.15)$$

n > 20 olduğunda toplam koşu sayısı (a) normal dağılıma $N(m_a,)$ yaklaştığından bağımsızlığın testi için standart normal dağılımdan yararlanılabilir. Bu koşullarda test istatistiği olan Z aşağıdaki gibi tanımlanmaktadır.

$$Z = \frac{a-\mu_a}{\sigma_a} \qquad (4.16)$$

Denklem (4.14) ve (4.15)'ten elde edilen değerler denklem (4.16)'da yerine konularak,

$$Z = \frac{a-\left(\frac{2n-1}{3}\right)}{\sqrt{\frac{16n-29}{90}}} \qquad (4.17)$$

elde edilir. Hesaplanan test istatistiği $Z < Z_{\alpha/2}$ ise bağımsızlık hipotezi reddedilemez.

Örnek

- Sayıs Dizisi

u_i	2	4	7	6	9	19	13	6	1	11
s_i		1	1	0	1	1	0	0	0	1

$$s_i = \begin{cases} 1 & u_i \le u_{i+1} \text{ ise} \\ 0 & u_i \ge u_{i+1} \text{ ise} \end{cases} \qquad (i = 1, 2, ..., n\text{-}1) \qquad (4.13)$$

H_0 : Sayılar rassaldır, H_1 : Sayılar rassal değildir

a: Yukarı ve aşağı doğru koşuların sayısı

$$E[a] = \mu_a = \frac{2n-1}{3} = (2*10-1)/3=19/3=6{,}33$$

$$\sigma_a^2 = \frac{16n-29}{90} = (16*10-29)/90 = 1{,}45 \qquad Z = \frac{a-\mu_a}{\sigma_a} \qquad Z = \frac{a-\left(\frac{2n-1}{3}\right)}{\sqrt{\frac{16n-29}{90}}} = (5-6{,}33)/1{,}20 = -1{,}108$$

$Z < Z_{\alpha/2}$ ise H_0 kabul.

$|-1{,}108| < 1{,}96$ H_0 kabul üretilen sayılar rassaldır.

Ortalamaya Göre Koşu Testi

Yukarı ve aşağı doğru koşu testinde kronolojik olarak sıralanan sayıların biribirlerine göre büyüklükleri dikkate alınırken, ortalamaya göre koşu testinde dikkate alınan sayıların ortalamaya göre durumlarıdır. Özetle yukarı ve aşağı doğru koşu testinde u_{i+1} yerini ortalamaya terk eder. Buna göre,

yazılır.

$$s_i = \begin{cases} 0 & u_i \leq \text{ortalama ise} \\ 1 & u_i \geq \text{ortalama ise} \end{cases} \qquad (i = 1, 2, ..., n) \qquad (4.18)$$

Ortalamaya Göre Koşu Testi -devam

n_1 ortalamadan büyük, n_2 ortalamadan küçük olan rassal sayıların sayısı ve b de toplam koşu sayısı olsun. Buna göre, ortalamaya göre koşu testinde toplam koşu sayısı en fazla ($n_1 + n_2$), en az 1 olabilir. n çok büyük olduğunda gerçekten bağımsız sayı dizisindeki toplam koşu sayısının (b) beklenen değeri ve varyansı aşağıdaki gibi hesaplanır.

$$E[b] = \mu_b = \frac{2n_1n_2}{n} + \frac{1}{2} \qquad (4.19)$$

$$\sigma_b^2 = \frac{2n_1n_2(2n_1n_2 - n)}{n^2(n-1)} \qquad (4.20)$$

n_1 veya n_2'den herhangi biri 20'den büyük olduğunda b normal dağılıma N(m_b,) yaklaşır. Bu durumda bağımsızlığın test edilmesinde kullanılacak olan Z istatistiği aşağıdaki formül kullanılarak hesaplanır (Banks vd., 1996, s.306).

$$Z = \frac{b - \frac{2n_1n_2}{n} - \frac{1}{2}}{\sqrt{\frac{2n_1n_2(2n_1n_2 - n)}{n^2(n-1)}}} \qquad (4.21)$$

Aşağı ve yukarı doğru koşu testinde olduğu gibi $Z_{\alpha/2} < Z$ ise H_0 hipotezi reddedilerek sayıların bağımsız olmadığına karar verilir.

Ortalamaya göre yapılan koşu testinde ortalama yerine medyan konularak medyana göre koşu testi gerçekleştirilebilir.

Örnek

- Sayıs Dizisi

u_i	2	4	7	6	9	19	13	6	1	11

$$s_i = \begin{cases} 0 & u_i \leq \text{ortalama ise} \\ 1 & u_i \geq \text{ortalama ise} \end{cases}$$

Ortalama 7,8

s_i	0	0	0	0	1	1	1	0	0	1

H_0 : Sayılar rassaldır, H_1 : Sayılar rassal değildir

n_1 ortalamadan büyük, rassal sayıların sayısı

n_2 ortalamadan küçük olan rassal sayıların sayısı

b toplam koşu sayısı

$$E[b] = \mu_b = \frac{2n_1n_2}{n} + \frac{1}{2} = (2.4.6)/10 + 0,5 = 4,8 + 0,5 = 5,3$$

$$\sigma_b^2 = \frac{2n_1n_2(2n_1n_2 - n)}{n^2(n-1)} = 48(48-10)/100(9) = 1824/900 = 2,02$$

$$Z = \frac{b - \frac{2n_1n_2}{n} - \frac{1}{2}}{\sqrt{\frac{2n_1n_2(2n_1n_2 - n)}{n^2(n-1)}}} = -0,91$$

$Z < Z_{\alpha/2}$ ise H_0 kabul.

|-0,91| <1,96 H_0 kabul üretilen sayılar rassaldır.

Koşu Uzunluğu Testi

Koşularla ilgili bir başka test koşu uzunluğu testidir. Knuth'a göre (1981, ss.1-170) koşu uzunluğu testi en iyi testlerden biridir. Birçok bağımsızlık testini başarıyla geçen sahte rassal sayılar koşu uzunluğu testini geçememektedir. Bu nedenle diğer testleri uygulamak yerine koşu uzunluğu testini uygulamak araştırmacının yararına olur.

Dikkate alınan koşu uzunluğu k olmak üzere, uzunluğu k olan koşuların sayısı R_k olsun. Rassal sayıların bağımsız olması durumunda toplam yukarı ve aşağı doğru koşular için R_k'nın beklenen değeri aşağıdaki gibi hesaplanır.

$$k \leq n - 2 \text{ için } E[R_k] = \frac{2[n(k^2 + 3k + 1) - (k^3 + 3k^2 - k - 4)]}{(k+3)!} \qquad (4.22)$$

$$k = n - 1 \text{ için } E[R_k] = \frac{2}{n!} \qquad (4.23)$$

Benzer yaklaşımla ortalamanın altında ve üstündeki koşular için R_k'nın beklenen değeri

$$E(R_k) = \frac{nw_k}{E(K)} \qquad n > 20 \qquad (4.24)$$

formülü kullanılarak hesaplanır.

Koşu Uzunluğu Testi

Burada w_k koşu uzunluğunun k'ya eşit olmasının yaklaşık olasılığıdır. Bu olasılık aşağıdaki formülle hesaplanır.

$$w_k = \left(\frac{n_1}{n}\right)^k \left(\frac{n_2}{n}\right) + \left(\frac{n_1}{n}\right)\left(\frac{n_2}{n}\right)^k \qquad n > 20 \qquad (4.25)$$

Denklem (4.24)'ün paydasında bulunan E(K) koşunun yaklaşık olarak beklenen uzunluk değeridir. Beklenen uzunluk aşağıdaki gibi hesaplanır.

$$E(K) = \frac{n_1}{n_2} + \frac{n_2}{n_1} \qquad n > 20 \qquad (4.26)$$

Tüm uzunluklar için koşuların toplam sayısının yaklaşık beklenen değeri,

$$E(A) = \frac{n}{E(K)} \qquad n > 20 \qquad (4.27)$$

olur. Uzunluğu k olan koşuların gözlenen frekansları (f_k) ile teorik frekanslarının $E(R_k)$ karşılaştırılması için χ^2 testi uygulanır. χ^2 test istatistiği aşağıdaki eşitlikten hesaplanır.

$$\chi^2 = \sum_{k=1}^{L} \frac{[f_i - E(R_k)]^2}{E(R_k)} \qquad (4.28)$$

Formüldeki L; yukarı ve aşağı doğru koşu testi için (n – 1), ortalamaya göre koşu testi için n'ye eşittir. $\chi^2_{a,L-1} < \chi^2$ ise rassal sayıların bağımsız olduğu görüşü reddedilir.

Örnek

- Sayıs Dizisi

u_i	2	4	7	6	9	19	13	6	1	11
s_i		1	1	0	1	1	0	0	0	1

$$s_i = \begin{cases} 1 & u_i \le u_{i+1} \text{ ise} \\ 0 & u_i \ge u_{i+1} \text{ ise} \end{cases}$$

k: Dikkate alınan koşu uzunluğu

R_k:uzunluğu k (2) olan koşuların sayısı

Rassal sayıların bağımsız olması durumunda toplam yukarı ve aşağı doğru koşular için R_k'nın beklenen değeri aşağıdaki gibi hesaplanır.

$$k \le n - 2 \text{ için } E[R_k] = \frac{2[n(k^2 + 3k + 1) - (k^3 + 3k^2 - k - 4)]}{(k+3)!} = \frac{2[10(2^2 + 3.2 + 1) - (2^3 + 3.2^2 - 2 - 4)]}{(2+3)!} = 16$$

$$\chi^2 = \sum_{k=1}^{L} \frac{[f_i - E(R_k)]^2}{E(R_k)}$$

Formüldeki L; yukarı ve aşağı doğru koşu testi için (n – 1), ortalamaya göre koşu testi için n'ye eşittir. $\chi^2_{a,L-1} < \chi^2$ ise rassal sayıların bağımsız olduğu görüşü reddedilir.

Uygunluk Testleri

- Gözlemler sonucunda ulaşılan frekans dağılımının, teorik ya da deneysel varsayımlar çerçevesinde belirlenen fonksiyonlara uyup uymadığının belirlenmesi amacıyla gerçekleştirilen testlere uygunluk veya uyum iyiliği testleri denir. Diğer bir ifade ile uyum iyiliği testleri, anakütle yapısı bilinmeyen bir değişkenin gözlenen değerlerinin belirli bir olasılık fonksiyonuna ne kadar iyi uyduğunu belirlemek için kullanılır. Uyum iyiliği için önerilmiş olan birçok test vardır. Bunlardan en çok kullanılanları ki-kare, Kolmogorov-Smirnov, Anderson-Darling ve Cramer-von Mises testleridir. Gözlenen değerlerin belirli bir teorik dağılıma uygunluğu için tek bir testin kullanılması yeterli olmayabilir. Bu konuda alınan karara duyulan güveni artırmak için yukarıda belirtilen her bir testin uygulanması daha doğru olur.

Ki-kare Testi

Uyum iyiliğinin testinde kullanılan ki-kare testi en eski testlerden biridir. İstatistik disiplininin önde gelen isimleri arasında kabul edilen Karl Pearson tarafından geliştirilmiştir (Gamgam, 1989, s.88). Yukarıda açıklandığı gibi örnek değerlerinin bir teorik dağılıma uygunluğunun test edilmesinde kullanılır.

n birimli ve m sınıflı (gruplu) bir örneğin frekans dağılımı bir teorik dağılım fonksiyonundan elde edilen teorik frekans dağılımının bir alt örneği ise, gözlenen (f_i) ve teorik (e_i) olmak üzere iki frekans dağılımı vardır. Bu varsayım geçerli ise frekanslar arasındaki farklılık 1 - α sınırları içinde kalacaktır. k, teorik dağılımın parametre sayısı olmak üzere, f_i ile e_i farklarının karelerinin teorik değerlere oranının toplamı, m büyüdükçe, m – k - 1 serbestlik dereceli teorik χ^2 dağılımına yaklaşan bir test istatistiği vermektedir.

$$\chi^2 = \sum_{i=1}^{m} \frac{(f_i - e_i)^2}{e_i} \qquad (4.63)$$

Bu istatistik m – k - 1 serbestlik dereceli χ^2 dağılımına uyduğundan test istatistiği, $\chi^2_{\alpha,sd}$ tablo değeri ile karşılaştırılarak frekans dağılımının teorik dağılıma uygunluğu araştırılır. Eğer tablo değeri hesap değerinden küçük ise H_0 red edilir.

Bu testte dikkat edilmesi gereken şey, her sınıfın teorik (beklenen) ve gözlenen frekanslarının 5'ten küçük olmamasıdır. Böyle bir durumla karşılaşıldığında sınıfların birleştirilmesi yoluna gidilir. Ancak örnek sayısı az olduğunda bunu gerçekleştirmek mümkün olmayabilir.

Örnek

- On yüzlü bir zar 100 defa atılmıştır. Bu atışlarda gelen sayılar aşağıda verilmiştir. Gelen sayılar uniform dağılmaktamıdır, test edinz. (α = 0,05)

	0	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**Toplam**
Gerçekleşen	11	13	14	12	9	10	12	6	5	8	100
Beklenen	10	10	10	10	10	10	10	10	10	10	100

$$\chi^2 = \sum_{i=1}^{m} \frac{(f_i - e_i)^2}{e_i} = \frac{(11-10)^2}{10} + \frac{(13-10)^2}{10} + \ldots + \frac{(8-10)^2}{10} = \frac{80}{10} = 8$$

- H_0 : Veriler uniform dağılmaktadır, H_1 : Veriler uniformdağılmamaktadır
- $\chi^2_{\alpha,sd} < \chi^2$ hesap ise H_0 red.
- $\chi^2_{0,05,9}$ =16,92> χ^2 =8 H_0 kabul.

Kolmogorov-Smirnov Testi

Örnek dağılımının belirli bir teorik dağılıma uygunluğunu sınamada başvurulabilecek parametrik olmayan testlerden biri de Kolmogorov-Smirnov (K-S) testidir. İlk olarak Rus matematikçi A. N. Kolmogorov tarafından 1933'te ortaya atılan bu test 1939'da yine bir Rus matematikçi olan N. V. Smirnov tarafından iki rassal değişkenin uyum iyiliği için geliştirilmiştir. Uygulamacıya χ^2 testinden daha fazla kolaylık sağlayan bu testte, verilerin gruplanmasına gerek olmadığından bilgi kaybı söz konusu olmaz ve sınıf aralığı belirleme sorunu yaşanmaz. Bununla birlikte, χ^2 testine göre bazı dezavantajları da vardır. Kesikli dağılımlar için gerekli olan kritik değerler hazırda bulunmadığından ve bunların karmaşık formüllerle hesaplanmaları gerektiğinden, K-S testi genelde sürekli dağılımlar için kullanılır (Gleser, 1985). Diğer taraftan K-S testinin orijinal formu test edilen teorik dağılımın tüm parametrelerinin bilinmesini gerektirir (Law ve Kelton, 1991, s.388). Parametrelerin tamamı bilinmiyorsa K-S testinin en iyi alternatifi Lilifors testidir (Daniel, 1990, s.327).

K-S testinde biri gözlenen değerlerin diğeri teorik dağılımın birikimli dağılım fonksiyonu olmak üzere iki fonksiyon üzerinde durulur. Gözlenen frekansların birikimli dağılımı ile teorik dağılımdan bulunan birikimli dağılım karşılaştırılır. İki dağılımın birbirlerine en uzak oldukları noktadaki fark hesaplanır ve bu fark K-S tablosundan okunan kritik değerle karşılaştırılır.

Kolmogorov-Smirnov Testi-devam

- Örnekten elde edilen birikimli frekans dağılımı $F_n(x)$ ve teorik dağılım fonksiyonu F(x) olmak üzere test istatistiği,
- D = enb{$|F_n(x) - F(x)|$}   (4.64)
- formülüyle hesaplanır. D değeri grafiksel gösterimde $F_n(x)$ ile F(x) arasındaki en büyük dikey uzaklıktır. α hata payı için hesaplanan D değeri, K-S tablo değeri ile $D_{n,\alpha}$ karşılaştırılır. $D > D_{n,\alpha}$ ise, örnek dağılımının, önerilen teorik dağılımdan farklı olduğu, bir başka ifade ile $F_n(x) \neq F(x)$ olduğu kararlaştırılır.
- Hem K-S hem de ki-kare testleri örneğin yeterince büyük olmasını gerektiren testlerdir. Ki-kare testinin uygulanabilmesi için gereken örnek büyüklüğü (en az 250), K-S testi için gerekenden daha fazladır.

Örnek

- On yüzlü bir zar 100 defa atılmıştır. Bu atışlarda gelen sayılar aşağıda verilmiştir. Gelen sayılar uniform dağılmaktamıdır, test edinz.

	0	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**Toplam**
Gerçekleşen	11	13	14	12	9	10	12	6	5	8	100
Beklenen	10	10	10	10	10	10	10	10	10	10	100
$f_n(x)$	0,11	0,13	0,14	0,12	0,09	0,10	0,12	0,06	0,05	0,08	1
f(x)	0,10	0,10	0,10	0,10	0,10	0,10	0,10	0,10	0,10	0,10	1
$F_n(x)$	0,11	0,24	0,38	0,50	0,59	0,69	0,81	0,87	0,92	1	
F(x)	0,10	0,20	0,30	0,40	0,50	0,60	0,70	0,80	0,90	1	
$\|F_n(x) - F(x)\|$	0,01	0,04	0,08	0,10	0,09	0,09	0,11	0,07	0,02	0	

- H_0 : Veriler uniform dağılmaktadır, H_1 : Veriler uniform dağılmamaktadır
- $D > D_{n,\alpha}$ hesap ise H_0 red. D =0,11 < $D_{n,\alpha}$ =1,36/√n= 1,36/√100 H_0 kabul.

Anderson-Darling Testi

Tüm x değerleri için $|F_n(x) – F(x)|$ farkını aynı ağırlıkla değerlendirmesi K-S testinin önemli bir eksikliğidir. Oysa dağılımların büyük bir bölümü esas olarak kuyruk kısımlarında farklıdır. İşte K-S'un yetersizliği tam da bu noktada ortaya çıkar ve farkın önemini belirlemede yetersiz kalır. Anderson-Darling (1954) dağılımların kuyruklarındaki farklılıkların belirlenmesine yönelik bir test geliştirmişlerdir. Anderson-Darling (A-D) testi olarak bilinen test, K-S testinden daha güçlüdür ve normal, üstel, Weibull gibi birçok dağılım için kullanılabilmektedir

Anderson-Darling Testi-devam

A_n^2 ile gösterilen A-D test istatistiği aşağıdaki formülle hesaplanır.

$$A_n^2 = n\int_{-\infty}^{\infty}[F_n(x) - F(x)]^2\psi(x)f(x)dx \qquad (4.65)$$

Burada ağırlık fonksiyonu olan $\psi(x)$ aşağıdaki gibi tanımlanmıştır.

$$\psi(x) = \frac{1}{F_n(x)[1 - F(x)]} \qquad (4.66)$$

Görüldüğü gibi yalnızca $[F_n(x) - F(x)]$ farklarının karelerinin ağırlıklı ortalamasıdır. F(x)'in 0'a (sol kuyruk) ve 1'e (sağ kuyruk) yakın değerleri için ağırlıklar en büyük olur. $Z_i = F(x_i)$ (i = 1, 2, ..., n) olarak tanımlandığında,

$$A_n^2 = \frac{-\left[\sum_{i=1}^{n}(2i-1)[\ln Z_i + \ln(1 - Z_{n+1-i})]\right]}{n} - n \qquad (4.67)$$

olduğu gösterilebilir (Law ve Kelton, 1991, s.392). ağırlıklı uzaklık olduğundan > $a_{n,1-\alpha}$ ise dağılımların kuyrukları arasında istatistiksel bakımdan fark yoktur hipotezi reddedilir. Eşitsizliğin sağ tarafında bulunan $a_{n,1-\alpha}$ kritik değeri uygun tablolardan bulunabilir.

Örnek

Cramer – von Mises Testi

- Uyum iyiliği testlerinden biri de Cramer ve von Mises tarafından 1928 yılında ortaya atılan Cramer-von Mises testidir. Yalnızca frekans ve teorik dağılımlar arasındaki en büyük dikey farkları değil, aynı zamanda n örnek büyüklüğü için n tane farkı da ele alır (Kıroğlu, 1996, s.38). Varsayımları K-S testinin varsayımları ile aynı olan bu testte de genellikle iki yanlı hipotezler kurulur.
- Test için öncelikle rassal değişkenin gözlenen değerleri $x_1 < x_2 < ... < x_n$ şeklinde küçükten büyüğe doğru sıralanır. i sıra indisi ve F(x) teorik dağılım fonksiyonu olmak üzere test istatistiği aşağıdaki formülle hesaplanır.

$$T_3 = \frac{1}{12n} + \sum_{i=1}^{n}\left(F(x) - \frac{2i-1}{2n}\right)^2 \qquad (4.68)$$

- T_3 için hesaplanan değer, T_3'ün asimtotik dağılım fonksiyonlarından bulunan Wp tablo değeri ile karşılaştırılır. $T_3 < W_p$ ise $F_n(x) = F(x)$ şeklinde kurulan H_o hipotezi red edilir (Kıroğlu, 1996, s.38).

Örnek

Rassal Değişken Türetme Yöntemleri

- Rassal sayıların üretililip test edilmesinden sonra, bu sayılardan hareketle belirli dağılımlara uygun rassal değişken türetim aşamasına geçilir. Rassal değişken türetilmesinde kullanılmak amacıyla önerilmiş birçok yöntem vardır. Rassal değişkenlerin belirli bir dağılıma uygun olarak türetilmesi için uygun yöntemin seçimi son derece önemlidir. Çok sayıda olan bu yöntemler genel olarak iki grupta toplanır: Tüm dağılımlar için uygulanabilecek yöntemler birinci grupta, çok özel ve uygulamada çok sık karşılaşılmayan yöntemler ikinci grupta toplanır (Pidd, 1990, s.182). Biz birinci grubu oluşturan yöntemler üzerinde duracağız.
- **Ters Dönüşüm Yöntemi**
- **Red-Kabul Yöntemi**
- **Bileşim Yöntemi**
- **Konvolüsyon Yöntemi**
- **Box-Muller Yöntemi**
- **Box-Muller Kutupsal Yöntemi**

Ters Dönüşüm Yöntemi

- Hem sürekli, hem de kesikli rassal değişken türetiminde kullanılabilen genel bir yöntemdir. Simülasyonda sıkça karşılaşılan tekdüze, üstel, üçgensel ve Weibull başta olmak üzere pek çok dağılıma uygun rassal değişken türetiminde rahatlıkla kullanılabilir. Çalışmanın bu kesiminde yöntem önce kesikli rassal değişkenler için açıklanacak daha sonra sürekli rassal değişkenler için kullanımın genel formu verilecektir.
- Olasılık yoğunluk fonksiyonu,
- P{X = x_j} = p_j (j = 0, 1, ... ve p_j = 1)(4.37)
- olan kesikli rassal değişken için rassal değer türetmek istenmesi durumunda, aşağıdaki yöntemin izlenmesi uygun olur. Öncelikle (0, 1) aralığında tekdüze dağılan sayılar üretilir. Üretilen bu sayıların her biri aşağıdaki kontrolden geçirilir.

$$X = \begin{cases} x_0 & U < p_0 \text{ için} \\ x_1 & p_0 \leq U < p_0 + p_1 \text{ için} \\ . \quad . & . \\ . \quad . & . \\ x_j & \sum_{i=1}^{j-1} p_i \leq U < \sum_{i=1}^{j} p_i \text{ için} \\ . \quad . & . \\ . \quad . & . \end{cases} \qquad (4.38)$$

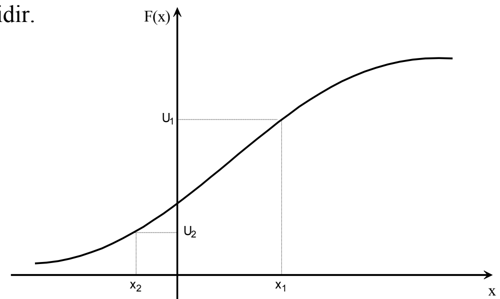
Ters Dönüşüm Yöntemi-devam

- 0 < a < b için P{a ≤ U < b} = b - a olduğundan,
- (4.39)
- yazılabilir. Bu istenilen dağılıma uygun biçimde dağılan X'e ulaşıldığı anlamına gelir. Ters dönüşüm yöntemi bir algoritma olarak aşağıdaki gibi açıklanabilir (Ross, 1990, s. 45).
- **1. Adım**: U(0, 1)'e uygun rassal bir sayı U üretilmesi,
- U < p_0 ise, X = x_0 saptamasının gerçekleştirilmesi ve durulması,
- U < $p_0 + p_1$ ise, X = x_1 saptamasının gerçekleştirilmesi ve durulması,
- U < $p_0 + p_1 + p_2$ ise, X = x_2 saptamasının gerçekleştirilmesi ve durulması,
- . . .
- U < $p_0 + p_1 + p_2 + ... + p_k$ ise, X = x_k saptamasının gerçekleştirilmesi ve durulması.
- **Adım 2**: x_i'ler (i ≥ 0) $x_0 < x_1 < ...$ şeklinde sıralandığında ve X rassal değişkeninin dağılım fonksiyonu F olarak tanımlandığında,
- F(x_k) = (4.40)
- olur. Sonuçta, F(x_{j-1}) ≤ U ≤ F(x_j) ise X = x_j olduğu kararlaştırılır.

Ters Dönüşüm Yöntemi-devam

- Özetle, bir rassal sayının (U) üretilmesinden sonra U'nun bulunduğu ($F(x_{j-1})$, $F(x_j)$) aralığının belirlenmesi X'in değerine ulaşılmasını sağlar. Bu F(U)'nun tersinin bulunması anlamına geldiğinden bu yönteme kesikli ters dönüşüm yöntemi denir.
- Yukarıda açıklandığı gibi, ters dönüşüm yöntemi sürekli dağılımlarla da kullanılabilir. Ters dönüşüm yöntemi genellikle birikimli dağılım fonksiyonu kapalı form sergileyen sürekli dağılımlara uygulanır.
- U(0,1) dağılan rassal bir değişken olsun. Herhangi bir sürekli dağılım fonksiyonu F için $X = F^{-1}(U)$ şeklinde tanımlanan X rassal değişkeninin dağılımı F'tir . Sürekli bir rassal değişken (X) türetilmek istensin. $0 < F(x) < 1$ ise X'in dağılım fonksiyonu F, sürekli ve monoton artan bir fonksiyon olur. Bu, $x_1 < x_2$ ise $0 < F(x_1) \leq F(x_2) < 1$ olduğu anlamına gelir. F fonksiyonunun tersi F^{-1} olarak gösterildiğinde rassal değişkenin türetilmesinde kullanılan algoritma aşağıdaki gibidir.
- **Adım 1**: Rassal değişkenin birikimli dağılım fonksiyonu F(x)'in tanımlanması: F(x) ya temel bilgilerden hareketle doğrudan veya yoğunluk fonksiyonunun integralinin hesaplanmasının ardından belirlenir.
- **Adım 2**: Tekdüze dağılıma uygun bir rassal sayı (u) üretilmesi
- **Adım 3**: Birinci adımda belirlenen F(x) ile ikinci adımda belirlenen rassal sayının eşitlenmesi: F(x) = u ve eşitliğin x'e göre çözülmesi. Bu işlem rassal değişkenin ters dağılım fonksiyonunun belirlenmesini ve u'nun x'e dönüştürülmesini sağlar.

- Yöntem geometrik olarak da açıklanabilir. Geometrik açıklama için X'in birikimli dağılım fonksiyonu F(x) çizilir. U(0,1) dağılan rassal bir sayı üretilir. Üretilen sayı F(x) eksenine yerleştirilir ve F(x) eğrisini veya süreksizliğini kesene kadar bu noktadan sağa doğru bir doğru çizilir. Kesişim noktasının x eksenindeki izdüşümü f(x)' in rassal değeridir.



Şekil 4.3: Ters Dönüşüm Yöntemi

- Ters dönüşüm yönteminin uygulanabilmesi için olasılık yoğunluk fonksiyonu f(x)'in bilinmesi ve F(x)'in tersi alınabilir bir fonksiyon olması gerekir.

Örnek

- Aşağıda verilen deneysel dağılıma uygun olarak, rassal sayılardan ters dönüşüm yöntemini kullanarak rassal değişken türetiniz.

Getiri	Frekans	f(x)	F(x)
15	3	0,15	0,15
10	2	0,10	0,25
5	7	0,35	0,60
-2	5	0,25	0,85
-3	3	0,15	1

Rassal Sayılar
0,11
0,19
0,25
0,09
0,98
0,75
0,55
0,66
0,67
0,43
0,36
0,81

Çözüm

Frekans	f(x)	F(x)	Alt Değer	Üst Değer	Getiri	Rassal Sayı	Xi
3	0,15	0,15	0	0,15	15	0,11	
2	0,10	0,25	0,15	0,25	10	0,19	
7	0,35	0,60	0,25	0,60	5	0,25	
5	0,25	0,85	0,60	0,85	-2	0,09	
3	0,15	1	0,85	1	-3	0,98	
						0,75	
						0,55	
						0,66	
						0,67	
						0,43	
						0,36	
						0,81	
Excel İçin							
A	B	C	D	E	F	G	H

H2 Hücresine =vlookup(G2;$D$2:$F$6;3) veya =vlookup(G2;$D$2:$F$6;3) yazarak kolayca yapılabilir.

Red-Kabul Yöntemi

- Kuyruk modellerinde kullanılan Erlang, proje çizelgeleme ve geliştirme tekniği PERT'de kullanılan beta dağılımı dahil bazı önemli dağılımların birikimli dağılım fonksiyonları kapalı olmadığından, yani tersleri alınamadığından bu dağılımlara ters dönüşüm yöntemi uygulanamaz. Bu koşullarda akla ilk gelen yöntem red-kabul yöntemidir.
- Uygulaması oldukça kolay olan red-kabul yöntemi von Neuman (1951) tarafından geliştirilmiştir. X rassal değişkeni (a,b) gibi sınırlı bir aralıkta değişiyorsa f(x)'in bu aralıkta tanımlı olduğu dağılım için kullanılması uygundur. Bu yöntem hem kesikli rassal değişken, hem de sürekli rassal değişken türetilmesinde kullanılabilir. Burada red-kabul yönteminin önce kesikli sonra da sürekli rassal değişkenler için kullanımı anlatılacaktır.
- Uygun bir kesikli dağılımdan rassal değerler çekilerek bu tekniğe uygulanır ve rassal değişkenin kabul edilip edilemeyeceği test edilir. Olasılık fonksiyonu $\{p_j, j \geq 0\}$ olan değişkenden hareketle, olasılık fonksiyonu $\{q_j, j \geq 0\}$ olan rassal değişkenin türetilmesinde kullanılacak etkin bir yöntemin varolması durumunda öncelikle bu değişken (Y) türetilir. Daha sonra türetilen bu değişken değerleri p_Y/q_Y ile orantılı bir olasılıkla kabul edilir. c aşağıdaki eşitsizliği gerçekleyen sabit bir sayı olsun.
- $\frac{p_j}{q_j} \leq c$ tüm j'ler için ve $p_j \neq 0$ (4.41)

Red-Kabul Yöntemi-devam

- Olasılık fonksiyonu $p_j = P\{X = j\}$ olan X kesikli rassal değişkeninin red kabul yöntemiyle türetilme algoritması aşağıda verilmiştir (Ross, 1990, s.52).
- **Adım 1**: Olasılık yoğunluk fonksiyonu q_j olan Y'ye ait değerlerin türetilmesi,
- **Adım 2**: Rassal sayı U'nun üretilmesi,
- **Adım 3**: $U \leq p_Y/cq_Y$ ise, X = Y kabul edilir, eşitsizlik sağlanmıyorsa birinci adıma dönülür.
- **Sürekli değişken türetme süreci**: Olasılık fonksiyonu g(x) olan bir sürekli rassal değişkenin türetilmesinde kullanılacak etkin bir yöntemin varlığı durumunda, öncelikle yoğunluk fonksiyonu g olan Y rassal değişkeninin türetilmesi gerçekleştirilir. Bu aşamada türetilen değer f(Y)/g(Y) ile orantılı olan olasılıkla kabul edilerek f(x) olasılık fonksiyonuna sahip X'in türetilmesi tamamlanır. Şöyle ki; c sabit olmak üzere,
- $\frac{f(y)}{g(y)} \leq c$ tüm y'ler için (4.42)
- Red-kabul yöntemiyle sürekli rassal değişken türetmek için aşağıdaki algoritma kullanılır (Ross, 1990, s.63).
- **Adım 1**: g olasılık fonksiyonuna sahip Y'nin türetilmesi
- **Adım 2**: Rassal sayı U'nun üretilmesi
- **Adım 3**: $U \leq \frac{f(y)}{cg(y)}$ ise, X = Y kabul edilir, eşitsizlik sağlanmıyorsa birinci adıma dönülür.

Örnek