

İSTANBUL TEKNİK ÜNİVERSİTESİ ★ FEN BİLİMLERİ ENSTİTÜSÜ

**GENETİK ALGORİTMAYA DAYALI KİTLESEL BİREYSELLEŞTİRME
AMAÇLI KONUT TASARIM MODELİ**

**YÜKSEK LİSANS TEZİ
Özge GÜNGÖR**

Anabilim Dalı : Bilişim

Programı : Mimari Tasarımda Bilişim

HAZİRAN 2010

İSTANBUL TEKNİK ÜNİVERSİTESİ ★ FEN BİLİMLERİ ENSTİTÜSÜ

**GENETİK ALGORİTMAYA DAYALI KİTLESEL BİREYSELLEŞTİRME
AMAÇLI KONUT TASARIM MODELİ**

**YÜKSEK LİSANS TEZİ
Özge GÜNGÖR
(523071015)**

Tezin Enstitüye Verildiği Tarih : 07 Mayıs 2010

Tezin Savunulduğu Tarih: 07 Haziran 2010

**Tez Danışmanı : Prof. Dr. Gülen ÇAĞDAŞ (İTÜ)
Diğer Jüri Üyeleri: Doç. Dr. Sinan Mert ŞENER (İTÜ)
Doç. Dr. Birgül ÇOLAKOĞLU (YTÜ)**

HAZİRAN 2010

Anne ve Babama,

ÖNSÖZ

Bilgisayar teknolojileri destekli tasarım ve üretim sistemleri ile bir ürünün ilk taslak aşamasından ürünün geliştirilmesine ve üretiminden kullanıcıya ulaştırılmasına kadar çok çeşitli üretim yolları geliştirilmiştir. Kitlesel bireyselleştirme de bu üretim yöntemlerinden birisidir. Bu tez çalışmasında evrimsel algoritmaların doğadan esinli form arama-bulma çalışmalarından etkilenilerek, mimari tasarımda uzman olmayan kullanıcının konut tasarımına yardımcı olması hedeflenen, genetik algoritmaya dayalı kitlesel bireyselleştirme amaçlı bir model geliştirilmiştir.

Tezimin her aşamasında özverili bir şekilde bana yol gösteren ve farklı yaklaşımları ile tezin gelişimine katkıda bulunan değerli danışmanım Prof. Dr. Gülen Çağdaş'a içtenlikle teşekkür ederim.

Bilgisayar programlama konusundaki engin tecrübesiyle bana sürekli destek olan çok sevgili arkadaşım Telekomünikasyon Yüksek Müh. Ufuk Ülüğ'e de teşekkürlerimi bir borç bilirim.

Ayrıca sadece bu tez çalışması sırasında değil, tüm eğitim hayatım süresince sonsuz sabırları ve sevgileri ile hep yanımda olan anneme, babama ve kardeşime çok teşekkür ederim.

Haziran, 2010

Özge GÜNGÖR
Mimar

İÇİNDEKİLER

Sayfa

ÖNSÖZ.....	v
İÇİNDEKİLER	vii
KISALTMALAR	ix
ÇİZELGE LİSTESİ.....	xi
ŞEKİL LİSTESİ.....	xiii
ÖZET.....	xix
SUMMARY	xxi
1. GİRİŞ	1
2. SERİ ÜRETİM VE KİTLESEL BİREYSELLEŞTİRME.....	5
2.1 Seri Üretim	5
2.2 Kitlesele Bireyselleştirme	8
2.3 Kitlesele Bireyselleştirmenin Türleri	8
2.3.1 İşbirliği ile bireyselleştirme	8
2.3.2 Uyarlanmış bireyselleştirme	10
2.3.3 Modüler hale getirme	13
2.3.3.1 Bileşen paylaşırma modülasyonu	15
2.3.3.2 Bileşen değıştirme modülasyonu	16
2.3.3.3 Uyuma göre kesme modülasyonu	17
2.3.3.4 Karma modülasyon	19
2.3.3.5 Omurgaya bağı modülasyon	20
2.3.3.6 Bölümsel modülasyon	21
2.4 Seri Üretim ile Kitlesele Bireyselleştirmenin Kıyaslanması.....	21
2.5 Konut Tasarımında Kitlesele Bireyselleştirme	22
2.5.1 Konut projelerinde üretim yaklaşımları	23
2.5.2 Kitlesele bireyselleştirme yaklaşımıyla konut tasarımında bilişim teknolojilerinin rolü	29
3. KONUT TASARIMINDA KİTLESEL BİREYSELLEŞTİRME	
ÖRNEKLERİ.....	31
3.1 Get Physical.....	31
3.2 Dry-In House	34
3.3 Embriyolojik Ev	36
3.4 Malagueria Konutları	41
4. KİTLESEL BİREYSELLEŞTİRME AMAÇLI EVRİMSEL YAKLAŞIM ..	49
4.1 Evrimsel Bilgi İşleme.....	49
4.1.1 Evrim stratejileri	52
4.1.2 Evrimsel programlama.....	52
4.1.3 Genetik Programlama	54
4.1.4 Genetik Algoritmalar	56
4.1.4.1 Genetik Algoritmaların temel teoremi	57
4.1.4.2 Genetik Algoritma kavramları	59

4.1.4.3 Genetik Algoritma parametreleri	70
4.1.4.4 Genetik Algoritma ile örnek problem çözümü	72
4.2 Evrimsel Tasarım.....	78
4.2.1 Evrimsel tasarım en iyilemesi	80
4.2.2 Yaratıcı evrimsel tasarım	82
4.2.3 Evrimsel sanat	83
4.2.4 Evrimsel yapay hayat formları	86
4.3 Genetik Algoritmaların Mimari Tasarım Örneklerinde Kullanılması.....	89
4.3.1 Konut kat planı tasarım modeli 1	89
4.3.2 Konut kat planı tasarım modeli 2	91
4.3.3 Biçim evrimine dayalı apartman bloğu tasarımı	99
5. GENETİK ALGORİTMAYA DAYALI KİTLESEL BİREYSELLEŞTİRME	
AMAÇLI KONUT TASARIM MODELİ	111
5.1 Geliştirilen Modelin Amacı ve Tanıtımı	111
5.2 Modelin Algoritması	113
5.2.1 Genetik Algoritma ile oda biçimlerinin belirlenmesi.....	114
5.2.2 Konut kat planı şemasının oluşturulması	124
5.3 Modelin Kitlesel Bireyselleştirme Amaçlı Arayüz Tasarımı	128
5.3.1 Veri giriş arayüzleri.....	128
5.3.2 Seçim arayüzleri.....	131
5.3.3 Grafik arayüzü.....	132
5.4 Farklı Kullanıcılara Özgü Konut Tiplerinin Geliştirilmesi	134
6. SONUÇLAR	139
KAYNAKLAR.....	143

KISALTMALAR

GA	: Genetik Algoritmalar
CNC	: Bilgisayarlı Sayısal Kontrol (Computer Numerical Control)
CAD	: Bilgisayar Destekli Tasarım (Computer Aided Design)
MIT	: Massachusetts Teknoloji Enstitüsü
FEMA	: Federal Acil Durum Yönetim Ajansı
ABS	: Akrilonitril Bütadien Stiren
XPS	: Extrüde Polistiren Köpük
ES	: Evrim Stratejileri
EP	: Evrimsel Programlama
GP	: Genetik Programlama
DNA	: Deokrisibo Nükleik Asit
G	: Genotip
F	: Fenotip
AL	: Yapay Hayat (Artificial Life)
VRML	: Sanal Gerçeklik Modelleme Dili (Virtual Reality Modeling Language)

ÇİZELGE LİSTESİ

Sayfa

Çizelge 2.1 : Konutta standardizasyon ve bireyselleştirme düzeyleri (Noguchi & Hernandez, 2005).....	26
---	----

ŞEKİL LİSTESİ

Sayfa

Şekil 2.1 : Citrohan evi (http://obviousmag.org/archives/2004/11/casa_citraen.html).	6
Şekil 2.2 : Goldenhook örgü firmasının web sitesinin, kullanıcı seçimleri ile ilerleyen arayüzleri (http://www.goldenhook.fr).	9
Şekil 2.3 : Ayakkabı kişiselleştirme diyagramı (Boër & Dulio, 2007).	10
Şekil 2.4 : Marmol Radziner firması tarafından tasarlanan ev modeli görseli (www.marmolradzinerprefab.com).	12
Şekil 2.5 : Marmol Radziner firması tarafından tasarlanan ev modelinde kullanıcının sistem ile etkileşimli katalog üzerinden malzeme seçimini gösteren arayüzü (www.marmolradzinerprefab.com).	13
Şekil 2.6 : Ürün ve hizmetlerin kitlesel özel üretim yöntemi ile üretilebilmesi için altı modülasyon tipi (Ulrich ve Tung, 1991; pine, 1999).	14
Şekil 2.7 : Akustik duvar (http://www.archdaily.com/15400/acoustic-barrier-onl)..	15
Şekil 2.8 : Akustik duvar detayları (http://www.archdaily.com/15400/acoustic-barrier-onl).	16
Şekil 2.9 : Create-A-Book© örnek sayfa (http://www.createabook.com).....	17
Şekil 2.10 : Manilow evi, case study: Garofalo Architects, 2007.	18
Şekil 2.11 : Youbar müşterilerinin istekleri doğrultusunda karışımlar hazırlamaktadırlar (http://youbars.com).	20
Şekil 2.12 : Katılımcıların belirlediği konutların tasarımlarının maketi, İzmit yeni yerleşmeler projesi (Çavdar, 1978).	24
Şekil 2.13 : Yerlerine monte edilen döşeme ve kolonlar, İzmit yeni yerleşimler projesi (Bulca, 1977).	25
Şekil 2.14 : Açık kaynaklı yapı (Larson, 2004).	28
Şekil 3.1 : Get Physical arayüzü ekranı (http://www.physicaldesignco.com).....	31
Şekil 3.2 : Google Sketch-Up programı ile oluşturulmuş hazır model (http://www.physicaldesignco.com).....	32
Şekil 3.3 : Google Sketch-Up programı ile oluşturulmuş modeller kullanıcı tarafından kişiselleştirilmektedir (http://www.physicaldesignco.com)...	32
Şekil 3.4 : Birimlerin tasarım, dönüşüm, üretim ve birleştirme evreleri (www.physicaldesignco.com).	33
Şekil 3.5 : Malzemelerin üç boyutlu birleşim detayları (www.physicaldesignco.com).	33
Şekil 3.6 : Afet konutları ile Dry-In projesinin karşılaştırılması (http://ddbnola.org).	34
Şekil 3.7 : Dry-In ev projesi tasarım aşamaları (http://ddbnola.org).....	35
Şekil 3.8 : Dry-In ev projesi inşa edilen prototipi (http://ddbnola.org).	36
Şekil 3.9 : Embriyolojik ev projesinin topoğrafya ile ilişkisi (Kolarevic, 2003).	37
Şekil 3.10 : Embriyolojik ev projesi prototipleri (http://www.glform.com/).	38
Şekil 3.11 : Embriyolojik ev projesi alüminyum dış kabuk rengi seçimi (http://www.glform.com/).	39

Şekil 3.12 : Embriyolojik ev projesi gölgelendirme malzemesi seçimi (http://www.glform.com/).	39
Şekil 3.13 : Embriyolojik ev projesi korozyona maruz kalacak metallerin boya tercihlerinin girilmesi (http://www.glform.com/).	40
Şekil 3.14 : Embriyolojik ev projesinde interaktif ara yüzden cam kaplama renginin tercihi (http://www.glform.com/).	40
Şekil 3.15 : Embriyolojik ev projesinde sonuç formunun tercihi (http://www.glform.com/).	41
Şekil 3.16 : Embriyolojik ev projesinde XPS malzemenin CNC yardımı ile şekillenen prototip modelinin üretimi ve sergilenmesi (www.embryological.com).	41
Şekil 3.17 : Alvaro Siza tarafından 1977-1997 yılları arasında tasarlanan mevcut Malagueria konut tasarım tipleri (http://home.fa.utl.pt/~jduarte/malag/Malag/).	42
Şekil 3.18 : Alvaro Siza tarafından 1977-1997 yılları arasında inşa edilen Malagueria konut yerleşimi ve tasarım tiplerinin gösterimi (http://home.fa.utl.pt/~jduarte/malag/Malag/).	42
Şekil 3.19 : Diyagramlar Malagueira, konutlarında Alvaro Siza Veira'nın tasarım sistemine bir konut için uygulanan tek bir biçim grameri kuralının ürettiği farklı sonuçları göstermektedir (Duarte, 2005).	43
Şekil 3.20 : Malagueria konutlarında biçim grameri elemanları (http://home.fa.utl.pt/~jduarte/malag/Malag/).	44
Şekil 3.21 : Malagueria konutlarında uygulanan biçim grameri kurallarından bazıları (Duarte, 2005).	45
Şekil 3.22 : Malagueria konutlarında uygulanan biçim grameri ile kat planı şemaları üretiminden bir örnek (Duarte, 2005).	45
Şekil 3.23 : Malagueria konutlarında kullanıcı etkileşimli konut tasarımı (http://home.fa.utl.pt/~jduarte/malag/Malag/).	46
Şekil 3.24 : Malagueria konutlarında kullanıcı ile etkileşime geçen bilgi motoru (http://home.fa.utl.pt/~jduarte/malag/Malag/).	47
Şekil 4.1 : Örnek bir ev tasarımına arama uzayında parametrik değerlerle çözüm aranması (Bentley, 1999).	50
Şekil 4.2 : Popülasyon sayısı dört olarak evrimleşen konut tasarımlarının dört nesli (Gelecek neslin ebeveynleri yuvarlak içine alınanlardır) (Bentley, 1999).	51
Şekil 4.3 : Evrimleşen konut tasarımlarının her nesilde arama uzayındaki yerleri (Bentley, 1999).	51
Şekil 4.4 : Evrimsel programlamanın akış diyagramı (Bentley, 1999).	53
Şekil 4.5 : Ağaç kodlama ile kodlanmış kromozomların örneği (Obitko,1998).	55
Şekil 4.6 : Ağaç kodlamada ebeveyn kromozomların bir noktadan kesilerek çaprazlanması (Obitko, 1998).	55
Şekil 4.7 : Arama uzayındaki genotipler ile çözüm uzayındaki fenotiplerin eşleşmesi (Bentley, 1999).	60
Şekil 4.8 : Nesiller gelişikçe değişen uygunluk değeri (Bentley, 1999).	61
Şekil 4.9 : Genetik sürecin yaşam döngüsü	62
Şekil 4.10 : Rulet çemberi seçimi (Obitko, 1998).	65
Şekil 4.11 : Kromozomların uygunluk değer sırasına göre dizilimi (sıralamadan önce) (Obitko, 1998).	66
Şekil 4.12 : Kromozomların uygunluk değer sırasına göre dizilimi (sıralamadan sonra) (Obitko, 1998).	66

Şekil 4.13 : Basit çaprazlama operatörü (Bentley, 1999).	67
Şekil 4.14 : Çaprazlama yöntemleri ve etkileri, çaprazlama yapılanların altları çizilmiştir (Bolat, Erol ve İmrak, 2004).	68
Şekil 4.15 : Basit mutasyon operatörü (Bentley, 1999).	69
Şekil 4.16 : Mutasyon yöntemleri ve etkileri (Bolat, Erol ve İmrak, 2004).	69
Şekil 4.17 : Dikdörtgenlerin rastgele yerleşimi (Üçoluk, 2002).	73
Şekil 4.18 : Dikdörtgenlerin geçerli yerleşimi (Üçoluk, 2002).	73
Şekil 4.19 : Dikdörtgenlerin geçerli yerleşimi (Üçoluk, 2002).	73
Şekil 4.20 : Kromozomları oluşturan genlerin kodlanması (Üçoluk, 2002).	74
Şekil 4.21 : Belirlenmiş dizgeler havuzda tutulur (Üçoluk, 2002).	75
Şekil 4.22 : Çaprazlanmak üzere eşleştirilmiş iki kromozom (Üçoluk, 2002).	76
Şekil 4.23 : İki kromozomun çaprazlanmasında rastgele seçilmiş çaprazlama noktası (Üçoluk, 2002).	77
Şekil 4.24 : Çaprazlama sonrası oluşan yeni kromozomlar (Üçoluk, 2002).	77
Şekil 4.25 : Çaprazlama sonrası oluşan yeni kromozomların bit dizilimine göre aldığı değerler (Üçoluk, 2002).	77
Şekil 4.26 : Evrimsel bilgi işleme ve evrimsel tasarım, bilgisayar bilimleri ve evrimsel biyolojinin bir araya gelmesiyle ilişkilidir (Bentley, 1999).	79
Şekil 4.27 : Bilgisayarlı evrimsel tasarım görüşleri (Bentley, 1999).	79
Şekil 4.28 : Bir masanın evrimsel tasarım en iyilemesi (Bentley, 1999).	81
Şekil 4.29 : Yaratıcı evrimsel tasarım örneği süper yüzey (http://www.amandalevetearchitects.com/news/)	83
Şekil 4.30 : Rowbottom'un evrimsel sanat arayüz ekranı (Bentley, 1999).	84
Şekil 4.31 : NEvAr yazılımı kullanılarak üretilmiş bir evrimsel sanat örneği (Rooke, 1999).	84
Şekil 4.32 : Artistik masa evrimi (Bentley, 1999).	86
Şekil 4.33 : Sims'in evrimsel yapay hayat formu tasarımları (Bentley, 1999).	87
Şekil 4.34 : Evrimsel yapay hayat formu (Bentley, 1999).	88
Şekil 4.35 : Odaların verilen ebat ve tercih edilen biçimde, boşluklar, üst üste çakışmalar ve dışa taşmalar minimize edilerek bir sınıra oturtulması evrim stratejisi kullanılarak uygulanmıştır (Elezkurtaj & Franck, 1999).	90
Şekil 4.36 : Modüllerin (odaların bölümleri) seçilen komşuluk ilişkilerine göre belirli bir dış kontör içerisinde Genetik Algoritmalar ile yeniden düzenlenmesi (Elezkurtaj & Franck, 1999).	90
Şekil 4.37 : Kullanıcı arayüzü solda topolojiyi (ilişki bağlantılarını), ortada kat planı geometrisini, sağda ise seçilen komşuluk ilişkilerinin bağ ağırlıklarını anlatan topolojik matris gözükmektedir (Elezkurtaj & Franck, 1999).	91
Şekil 4.38 : Çok seviyeli birleşme ve çoğalma (Rosenman, 1996).	92
Şekil 4.39 : Odaları oluşturan birim boşluk (Rosenman, 1996).	93
Şekil 4.40 : Kullanıcı arayüzünde bölgelerin seçimi (Rosenman, 1996).	94
Şekil 4.41 : Kullanıcı arayüzünde yaşama bölgesinden oda seçimi (Rosenman, 1996).	95
Şekil 4.42 : Oturma odası için üretilen ilk popülasyon (Rosenman, 1996).	95
Şekil 4.43 : Nesilden bir odanın seçilmesi (Rosenman, 1996).	96
Şekil 4.44 : 38. nesilde ilk yürütmenin durdurulması sonucunda elde edilen odalar (Rosenman, 1996).	97
Şekil 4.45 : Mutfak ve giriş seçenekleri (Rosenman, 1996).	97
Şekil 4.46 : Yaşam bölgesinden üretilen ilk popülasyon (Rosenman, 1996).	98

Şekil 4.47 : Yaşam bölgesinden üretilen 1.yürütmedeki 10.popülasyon (Rosenman, 1996).....	99
Şekil 4.48 : Biçim grameri kuralları ile üretilen basit bir tasarım (Couchoulas, 2003).	101
Şekil 4.49 : 4m x 4m x 4m'lik yatay ve dikeyde çalışan sirkülasyon birimi (Couchoulas, 2003).	102
Şekil 4.50 : 16m x 4m x 4m'lik bir apartman birimi tasarımı (Couchoulas, 2003).	102
Şekil 4.51 : Apartman bloğu gramerinde birimlerin temsili ve oluşturulan ilk biçim (Couchoulas, 2003).	103
Şekil 4.52 : Apartman bloğu biçim gramerindeki üç ana kural (Couchoulas, 2003).	104
Şekil 4.53 : Apartman bloğu biçim gramerindeki bütün kurallar (Couchoulas, 2003).	104
Şekil 4.54 : Bir apartmanın biçim grameri kullanılarak geliştirilen örnek tasarım dizisi (Couchoulas, 2003).....	105
Şekil 4.55 : Apartmanın bloğu biçim gramerinde her kübik modülün numaralandırılması (Couchoulas, 2003).	105
Şekil 4.56 : Apartmanın bloğu biçim gramerinde bir örneğin değerlerinin dizi olarak gösterilmesi (Couchoulas, 2003).	106
Şekil 4.57 : VRML ile oluşturulan kullanıcı arayüzü (Couchoulas, 2003).	106
Şekil 4.58 : İsteklerin belirlendiği en iyileme arayüzü (Couchoulas, 2003).....	107
Şekil 4.59 : Uzunluk maksimum değerde olunca elde edilen çeşitli kule tasarımları (Couchoulas, 2003).	108
Şekil 4.60 : Aynı yürütmenin 459, 462 ve 461. nesillerinin en yüksek değerli tasarım örneklerinin birbirine yakınsaması (Couchoulas, 2003).	108
Şekil 4.61 : Biçim evrimi ile elde edilen geliştirilmiş konut bloğu tasarımı (Couchoulas, 2003).	109
Şekil 5.1 : Genetik algoritmaya dayalı kitlesel bireyselleştirme amaçlı modelin kullanıcı ile etkileşimli tasarımı.	112
Şekil 5.2 : Genetik algoritmaya dayalı kitlesel bireyselleştirme amaçlı modelin akış diyagramı.	114
Şekil 5.3 : Modelde oda biçimlerinin oluşturulmasına ilişkin akış diyagramı.	116
Şekil 5.4 : Bir oda oluşumunun fenotipler ve genotipleriyle açıklanması.	118
Şekil 5.5 : Bir çaprazlama işlemcisinin fonksiyonunun oda oluşturma örneği üzerinden gösterimi.	123
Şekil 5.6 : Genetik Algoritmalar ile üretilen 9 birim karelik çocuk popülasyon örnekleri.....	124
Şekil 5.7 : Odaların oluşturduğu kat planı şemasının akış diyagramı.	125
Şekil 5.8 : Arayüz üzerinden modele arazi özellikleri ve konut tipi bilgilerinin girilmesi.....	130
Şekil 5.9 : Arayüz üzerinden modele odaların olması istenilen topolojik özelliklerin girilmesi.....	130
Şekil 5.10 : Oda tasarımların arayüz üzerinden kontrol edilerek kullanıcının beğenisine göre seçimlerin yapılması.....	131
Şekil 5.11 : Oluşturulan kat planlarını arasından tercih yapılan seçim arayüzü.....	132
Şekil 5.12 : Tercih edilen kat planı tasarımını üç boyutlu olarak görselleştiren grafik arayüzü.	133
Şekil 5.13 : Üç boyutlu görselleştirmeyi sağlayan grafik arayüzü.	133
Şekil 5.14 : Bir aileye ait, üç katlı, sınırları belirli konut tasarımı perspektif 1.....	134
Şekil 5.15 : Bir aileye ait, üç katlı, sınırları belirli konut tasarımı perspektif 2.....	135

Şekil 5.16 : İki katlı, sınırları belirsiz, yazlık konut tasarımı perspektif 1	135
Şekil 5.17 : İki katlı, sınırları belirsiz, yazlık konut tasarımı perspektif 2	136
Şekil 5.18 : Farklı Kullanıcılara Ait Çok Katlı Konut Tasarımı Perspektif 1	136
Şekil 5.19 : Farklı Kullanıcılara Ait Çok Katlı Konut Tasarımı Perspektif 2	137

GENETİK ALGORİTMAYA DAYALI KİTLESEL BİREYSELLEŞTİRME AMAÇLI KONUT TASARIM MODELİ

ÖZET

Günümüzde seri üretim teknikleriyle üretilip pazarlanan ürünlerin yanı sıra, gelişen bilişim teknolojilerinin yadsınamaz etkisi ile ürünlerin tasarım ve üretiminde kullanıcıların da kısmen veya doğrudan söz sahibi olduğu bireyselleştirilmiş tasarımlar ön plana çıkmaktadır. Kullanıcının isteklerinin yön verdiği bu kişiye özel tasarım ve üretim anlayışı kitlesel bireyselleştirmez. Üretim öncesindeki tasarım, üretim ve sonrasında dağıtım gibi farklı seviyelerdeki kullanıcı ile iletişim ürünlerin kullanılabilirliği açısından büyük rol oynayan kitlesel bireyselleştirme, bilgisayar parçaları ve yazılımları, kişiye özel ayakkabı ve kıyafetler, otomotiv ürünleri gibi birçok farklı sektörde kendini göstermektedir. Kullanıcıların yaşam alışkanlıklarını ve tercihlerini en iyi yansıtan alan olan konutta kitlesel bireyselleştirme yavaş yavaş mimari tasarım anlayışına da yerleşmeye başlamıştır.

Bu tez çalışması kapsamında, konutların kullanıcı tercihleri doğrultusunda şekillenebilmesi ve mimar ile kullanıcı arasında istekleri yansıtan bir aracı olması açısından kitlesel bireyselleştirme amaçlı bir model tasarlanmıştır. Kullanıcı öncelikle arayüzler aracılığı ile modele konut yapısı ve oda tercihleri gibi isteklerini veri girişi olarak girmektedir. Modelde kullanıcıdan elde edilen verilerle oluşturulan çok sayıda oda seçeneği arasından, kullanıcının isteklerine en uygun olanlarının çoğalmasına izin veren ve uygun olmayan seçenekleri eleyen, doğadan esinli evrimsel bir yaklaşım olan Genetik Algoritmalar form arama ve kat planı üretme amaçlı kullanılmıştır. Konutun farklı katlarının üretiminden sonra oluşturulan kütle üç boyutlu olarak görselleştirilerek kullanıcıya mekânı deneyimleme fırsatı verilmektedir. Kullanıcı yönlendirmeleri doğrultusundaki verilerle oluşturulan model, iki veya üç boyutlu mimari çizim programlarında kullanılmak üzere bir ek yazılım ile dönüştürülerek, doğrudan mimara aktarılabilir. Ayrıca tez çalışması kapsamında oluşturulan bu model, ileride geliştirilerek farklı açılar ve eğrilerin de kullanıldığı daha esnek tasarımlar yapmaya olanak tanıyabilir.

A MASS CUSTOMIZATION ORIENTED HOUSING DESIGN MODEL BASED ON GENETIC ALGORITHM

SUMMARY

Today, right along with the products marketed and manufactured by the mass-production techniques, always developing informatics and technology have an undeniable impact on customized design, in which the users have a say on the products design and manufacture, is becoming more and more popular. This type of particular design and manufacture concept where the user's demands conduct all of the work is mass customization. Mass customization plays a very important role in the usability of the products as in different levels of the work like design, manufacture and distribution it is always in contact with the users. It is in fact standing out in many fields like computer parts and software, particular shoes and outfit, and also in the automotive industry. Mass customization is slowly settling down in architectural design concepts too like housing which is one of the best areas where users can reflect their living habits and preferences.

For housing being shaped according to the users preferences a tool which will reflect all the preferences is needed between the user and the architect so a mass customization oriented model has been designed in the context of this thesis. First of all with the intervention of user interface, the user enters the housing type and room like preferences the same way with the data entry. In the model, according to the data obtained by the users, among numerous room options chooses the ones that suit with the user preferences and permits them to proliferate and at the same time eliminates the non suitable ones, the nature inspired evolutionary approach Genetic Algorithms are used in the generation of form and floor plans. After obtaining the housing different floors plans and the three dimensional images of the generated mass are giving users the opportunity to experience the place. The created model can be transformed in an additional software program to two and three dimensional programs so that the user orientated design data can be transferred through a file to the architect. Besides, the model can be developed in the future so that it can use different angles and curves to allow more flexible designs.

1. GİRİŞ

Konut, insanların içgüdüsel olarak tasarlamaya ve inşa etmeye en eğilimli oldukları yapı birimidir. Bunun sebebi temel ihtiyaçların karşılandığı, kişilere özel alanları barındıran, kullanıcılarının kendilerini ait hissettiği mekanlar olmasıdır. Bireylerin yaşadıkları konuttan beklentileri, kişisel gereksinimlerinin ve isteklerinin karşılanarak, yaşam standartlarının korunması veya yükseltilmesidir. Konut üretimi yirminci yüzyıl başından, yaşadığımız zaman dilimine kadar giderek sanayi ürünleri ve araçlarının yardımı ile yapılmaktadır. Endüstrileşmiş ürünlerle ve seri üretim mantığı oluşturulmuş, birbirine benzer konutların içerisinde oturan kişi veya kişilerin her birinin kişisel ihtiyaçlarını tam olarak karşılayamaması önemli bir sorundur. Kullanıcı etkileşimli olarak şekillenmiş ihtiyaç programları doğrultusunda organize olan ve değişen tercihlere cevap verebilen konut tasarımı kullanıcıların memnuniyeti açısından kaçınılmaz olarak gereklidir.

Endüstriyel üretim yapılan her sektörde hissedilen bu kişiselleştirme eğilimi seri üretimden sonra yeni bir üretim anlayışı olan kitlesel bireyselleştirmeyi ortaya çıkarmıştır. Kitlesel bireyselleştirme, seri üretim verimliliğinde her kullanıcıya özel tasarlanmış ürünler sunmaktır (Piller & Kumar, 2006). Pine (1993), kitlesel bireyselleştirmenin doğru bir şekilde uygulanabilmesi için ileri teknoloji uygulamalarının, bilgisayarla bütünleşik imalat sistemlerinin, bilgisayar destekli tasarımın ve gelişmiş bilgisayar teknolojilerinin ön koşul olduğunu belirtir. Bilgisayar destekli tasarım sistemlerinin yazılım ve donanım desteği ile kitlesel bireyselleştirme, diğer alanlarda olduğu gibi konut tasarımında da kullanılmaya başlanmıştır. Konutta kitlesel bireyselleştirme anlayışı ile tasarımın, günümüzde sayılabilecek kadar sınırlı örnekler dışında, kullanıcıya sunulacak malzeme seçimi yapılan katalog yazılımlarının haricinde ilerleyememiş olduğu tespit edilmiştir. Bu tez çalışmasında kitlesel bireyselleştirme odaklı mimari tasarım probleminin, uzman tasarımcı olmayan kullanıcının istek ve seçimleri doğrultusunda, bilgisayar destekli bir yazılım aracılığı ile kat planı ve form tasarımı olarak çözümlenmesi araştırma konusu olarak seçilmiştir.

Kitlesel bireyselleştirme amaçlı konut tasarımını geliştirmek üzere, kullanıcıdan alınan bilgiler ile evrimsel bilgi işleme tekniğinin bir parçası olan Genetik Algoritmalar kullanılarak kat planı ve bina formu üreten, bir bilgisayar destekli tasarım modeli ortaya konulmuştur. Genetik Algoritmalar, ya da daha geniş anlamıyla Evrimsel Algoritmalar, doğadaki evrimsel süreçleri model olarak kullanan bilgisayara dayalı problem çözme teknikleridir. Geleneksel programlama teknikleriyle çözülmesi güç olan, özellikle sınıflandırma ve çok boyutlu optimizasyon problemleri bunların yardımıyla daha kolay ve hızlı olarak çözülebilmektedir (Atalağ, 2001).

Bu çalışma kapsamında Genetik Algoritmalar kullanılarak model oluşturulmasının temel sebebi, bir probleme çözüm aranırken skolastik düşünce yollarıyla elde edilemeyecek çok çeşitlilikte sonuç çözümleri ortaya koyabilmeleridir. Aynı durum yaratıcı tasarım sürecinde de geçerlidir. Yaratıcı tasarımda Genetik Algoritma teknikleri geleneksel olmayan yöntemle uygulanır, burada sistem için olası çözümler değil çözüme götüren araçlar bulunur, aynı zamanda yetersiz olan bireylerin yaşama şansı çok azaldığı için algoritmanın kötüye gidişi de zorlaşmaktadır. Genetik Algoritmaların böyle form üretme ve arama çalışmalarında kullanılması, geleneksel eskiz yöntemleri ile çalışan bir tasarımcının aynı sürede oluşturamayacağı kadar çok tasarım olasılığı arasından sürpriz olacak farklı sonuçlar ortaya çıkarabilmesi açısından da tercih edilmiştir.

Şüphesiz kitlesel bireyselleştirme amaçlı tasarlanan bu model aracılığı ile elde edilen kat planları ve bina formu, bir mimarın da tasarlayabileceği düzeyde olacaktır. Fakat bu tasarımları farklı kılan, bilgisayar modelindeki arayüz aracılığı ile mimar olmayan kullanıcının kendi istekleri doğrultusunda veri girişi yapıp, etkileşimli olarak beğenilerini modele aktararak kısa bir zaman diliminde sonuç ürün geliştirilmesidir. Bu yöntem ile elde edilen tasarımların, kullanıcı bilgi ve gereksinimlerine doğrudan ulaşıldığı için kullanışlılığının, kullanıcı istek ve ihtiyaçları genellenerek veya göz ardı edilerek tasarlanan konutlara göre daha yüksek olacağı düşünülmektedir.

Bu tez için geliştirilmiş Genetik Algoritmaya dayalı kitlesel bireyselleştirme amaçlı modelin tanıtılmasından önce modelin daha anlaşılır olması için kitlesel bireyselleştirme, kitlesel bireyselleştirmenin türleri ve konuttaki uygulama örnekleri ortaya konulacaktır. Ayrıca evrimsel bilgi işleme kavramı altında Genetik Algoritmalara genel olarak yer verilecektir. Bunun yanı sıra, literatürde yer alan

bilgisayarda Genetik Algoritmalar kullanılarak üretilen konut örneklerinin incelenmesi, ortaya konulacak modelin farklılıklarının görülebilmesi açısından yararlı olacaktır.

2. SERİ ÜRETİM VE KİTLESEL BİREYSELLEŞTİRME

Günümüzde bilgisayar teknolojilerinin de katkısı ile üretim anlayışı eskisine oranla gittikçe değişmektedir. 20. Yüzyılın başında otomotiv sektörünün gelişimi ile ortaya çıkan, aynı türden ürünü çok sayıda kimliği bilinmeyen muhtelif kullanıcı için hızlı ve büyük miktarlarda üretme mantığı olan seri üretim anlayışı, bulunduğumuz yüzyılda kullanıcıların farklı istekleri doğrultusunda ve belirli tüketicinin tercihinine yönelik olarak sınırlı miktarda üretim yapan kitlesel bireyselleştirmeye doğru yönelmektedir. Bu değişim tüm sektörlerde olduğu gibi konut tasarımında değişen kullanıcı isteklerinde de gözlemlenebilmektedir. Kullanıcılar kendi ihtiyaç ve istekleri doğrultusunda yapılan tasarımlarda yaşamayı, daha fazla kişinin genel ihtiyaçlarına yönelik üretilen ve çoğulcu tasarım anlayışı hakim olan konutlarda yaşamaya işlev ve estetik açısından tercih ederler.

Çok sayıda, standardize olmuş üretim yapan seri üretim ve her müşteri için özel üretim yapan kitlesel bireyselleştirme kavramları bu bölümde detaylı olarak açıklanacaktır.

2.1 Seri Üretim

Seri üretim; hareket eden bir montaj hattı ve yüksek oranda iş bölümüne dayalı, sermaye yoğun, üretim hattında çok basit işlemler yapan vasıfsız veya yarı vasıflı işgücü vasıtasıyla yüksek miktarlarda ve standart ürünlerin üretildiği bir üretim sistemidir (Bedir, 2002). Bu sisteme göre mümkün olduğunca en kısa sürede, en fazla standart parça üretimi aynı üretim hattında çalışan işçilerin aynı hareketi günde binlerce defa tekrarlamasıyla yapılmaktadır. Böylece büyük miktardaki işler, daha az bütçe ve daha kısa bir süreçte elde edilmektedir. 1910'larda Henry Ford'un girişimciliği ile tasarlanan Ford arabalarının T modelinin üretimi seri üretim anlayışını başlatmıştır. Henry Ford'un öncülük ettiği bu üretim modelinin başarısı ile farklı birçok endüstri sektörü de bu yönteme adapte olmaya başlamıştır.

Mimarlık ve inşaat sektöründe bu etkilenme ilk olarak 1920-1922 yılları arasında, Le Corbusier' in tasarladığı Citrohan evi modelinde görülmektedir (Şekil 2.1).

Citrohan evi bir ailenin yaşayabileceği bir prototip idi ve ismi Fransız otomobil markası Citroen'den türetilmişti. “Ev içinde yaşanan bir makinedir” (Courbusier, 1923) mantığı ile tıpkı arabalarda olduğu gibi evin standart parçalarının fabrikasyon olarak üretilip, kolayca birleştirilebilmesi ana fikirdi. Böylece herkes inşaat süreci kısa olan bu ekonomik evlerden edinebilecekti. O yıllarda deneyimsel olarak inşa edilen birkaç ev dışında mimari anlamda seri üretime geçilmiş sayılmaz. Fakat İkinci Dünya Savaşı'ndan sonra saldırılar sonucu yıkılmış Avrupa şehirlerinin yeniden üretimi için fabrikasyon ürünler kullanılmaya başlanmıştır. Prefabrike sistemler İngiltere, Fransa ve Almanya'da betonarme ve çelik malzemeye geliştirilirken, İsviçre'de ise ahşaba yönelme olmuştur (Arief & Burkhardt, 2002).



Şekil 2.1 : Citrohan evi (http://obviousmag.org/archives/2004/11/casa_citraen.html).

Prefabrik sistem ile fabrikalarda bir eve ait gerekli tüm yapı bileşenleri üretilebilmekteydi. Böylece insanların konuta olan ihtiyaçlarının artmasıyla seri sistemle elde edilen yapılaşma artmıştır. Ancak bu yolla bir fabrikadan üretilen mimari yapı elemanları yine aynı fabrikadan çıkan elemanlar ile eşleşebilmekte, diğer taraftan bir başka fabrikanın başka sistemiyle tasarlanan bileşenin kullanımını tamamen engellemekteydi. Bu nedenle kapalı sistem (closed system) ürünlerin kullanımı insanlar tarafından çok fazla tercih edilmemeye başlanmıştır.

Mimari anlamda seri üretim için bir başka bakış açısı olan parça kitleri (kit-of-parts) bina sistemlerini geliştirmişlerdir. Avrupa’da kabul edilen kapalı sistemler başka bileşenlerle uyarlanmayı engellemektedir, fakat Birleşmiş Devletlerdeki yüklenicilerin herhangi bir tedarikçiye bağlı olarak çalışmaktan kaçınma istekleri sayesinde teknoloji açık sistemlere (open systems) doğru yönelmiştir (Clark, 1986). Değişen isteklere göre uyarlanabilen bir yapay çevre için, açık sistemler teknik, organizasyonel ve finansal çözümlerle çok yönlü bir kavramdır. İnşa sürecinde kullanıcı katılımını, endüstrileşmeyi ve yeniden yapılandırmayı destekler (Cuperus, 2003). Bu gibi sistemler farklı tedarikçilerin mimari elemanlarının bir arada çalışabilmesini sağlıyordu. Böylece her şirket sadece bir bileşen ürüne yönelerek standart üretim yaptığı için maliyetler düşmüştü ve üretim süreci en iyileşmeye doğru gitmekteydi.

İlerleyen senelerde, bir araya gelip konut birimini oluşturan bileşenlerin sanayileşerek düşük bütçe ile üretilmesi ve seri üretimin gereği standardizasyon ile kaliteli ve kısa vadede üretilmesine rağmen, tasarımda oluşan konutlar tek tipleşmeye doğru dönüşmekteydi. Bu durumda tasarımcılar kullanıcı isteklerini ve katılımını da içeren yapı modelleri üzerinde çalışmaya başladılar. Bu amaçla 1972 de John Habraken teorilerini kapsayan bir kitap yayınlamıştır. Kitapta ‘destek’ (support) ve ‘doldurma’ (infill) olmak üzere iki ayrı terim üzerinde durmuştur. Destek, binanın taşıyıcı sistemini de içeren değişmeyen temel elemanlarını, doldurma ise kullanıcılar tarafından daha sonradan değiştirilebilen esnek elemanlarını tanımlamaktadır (Habraken, 1972) . Bu teoriye göre yapı gridlere bölünerek üç farklı fonksiyonel bölümden oluşmaktaydı, iki yanlarda ana odalar ve ortada da bir adet servis odası cepheye paralel olmakta ve bunların her biri sirkülasyon alanları veya verandalar ile sınırlandırılmaktaydı.

Belli bir süre sonra, Henry Ford’un seri üretim anlayışı kişisel tüketim mantığının gelişmesiyle gözden düşmeye başladı. Otomotiv sektöründe seri üretim yerine, kullanıcıların isteklerine göre üretimde küçük değişiklikler yapılarak üretilen yalın üretim (lean production) mantığı gelmeye başladı. Böylece kullanıcı seçtiği modeli, kendi istekleri doğrultusunda farklılaştırarak satın alabilmekteydi. Bu yeni sistem ile hem kullanıcının istediği model hızlıca üretilabiliyor, hem de stok sorunu çoğunlukla ortadan kalkıyordu.

2.2 Kitlesele bireyselleştirme

Yalın üretimin gelişimiyle yavaş yavaş başka bir üretim örneğine doğru gidilmekteydi. Bu yeni üretim modeli ‘kitlesele bireyselleştirme’ (Mass Customization) (Davis, 1987). Yirminci yüzyıl ekonomisinin “Fordian” paradigması ötesi olan kitlesele bireyselleştirme, Joseph Pine tarafından bağımsızca bireyselleştirilmiş seri üretim ürünleri ve servisleri olarak tanımlanmış ve böylece çeşitliliğin artması ve maliyetin bireyselleştirme ile doğru orantılı olarak artmaması şeklinde açıklanmıştır (Pine, 1993). Kitlesele bireyselleştirme, bilgisayar sistemlerinin, esnek üretim sistemleriyle bütünleştirilmesi sonucunda her bir müşteri için farklı bir ürünün oldukça büyük sayıda üretilmesini ifade etmektedir (Bardakçı, 2004). Kullanıcı birden çok sayıda tüketici için, çok sayıda üretilmiş ürünlerden herhangi bir tanesine değil; kendi ihtiyaç ve istekleri doğrultusunda, üretim aşamasından önce veya montaj aşamasında bireyselleştirmiş ürüne sahip olabilmektedir.

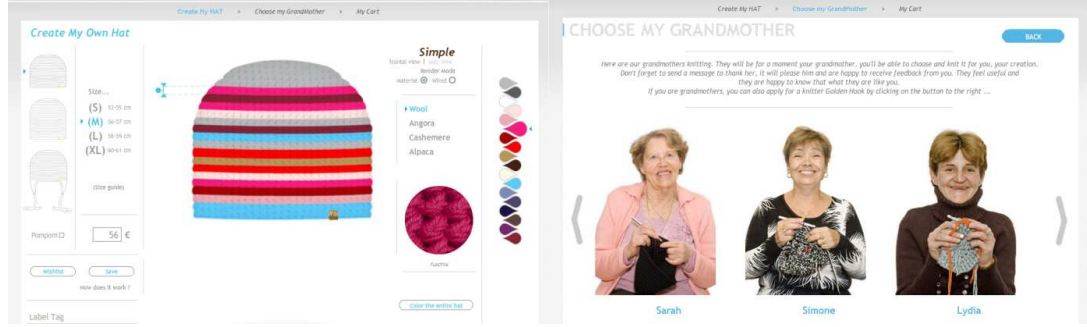
2.3 Kitlesele bireyselleştirmenin Türleri

Kitlesele bireyselleştirme hala gelişmekte olan bir kavramdır ve bu kavramın daha iyi anlaşılıp uygulanabilmesi için farklı yazarlar tarafından farklı tanımlamalar ve türler geliştirilmiştir. Bu tanımlamalardan bazıları işbirliği ile bireyselleştirme, uyarlanmış bireyselleştirme ve modüler hale getirmedir.

2.3.1 İşbirliği ile bireyselleştirme

Gilmore ve Pine, “Four Faces of Mass Customization” (1997) adlı makalelerinde kitlesele bireyselleştirmenin uygulanabilmesi için işbirliği ile bireyselleştirme ve uyarlanmış bireyselleştirme yönteminden bahsetmişlerdir. İşbirliği ile bireyselleştirme (Collaborative Customization) müşteri ile üretici firma arasındaki müşterinin istek ve ihtiyaçlarını tanımlayarak bu tanımlar doğrultusunda en doğru ürünü elde etmeye dayanan müşteriyle birebir çalışma yöntemidir. Bu yöntemde tüketici sonuç olarak istediği ürünün nasıl olacağını tam olarak bilmesede isteklerini paylaşarak, firmadan ihtiyaçlarına yönelik ürünü elde edebilir. Firma, stoklarında hiç olmayan bir ürünü kullanıcı için üretebilir veya ellerinde bulunan bir ürünü uyarlayarak müşterisine sunabilir; ama çoğunlukla bu yöntemde bitmiş bir ürün mevcut değildir. İşbirliği ile bireyselleştirme yöntemine basit bir örnek olarak,

sadece internet üzerinden satış yapan örgü şapka - atkı markası Goldenhook verilebilir.

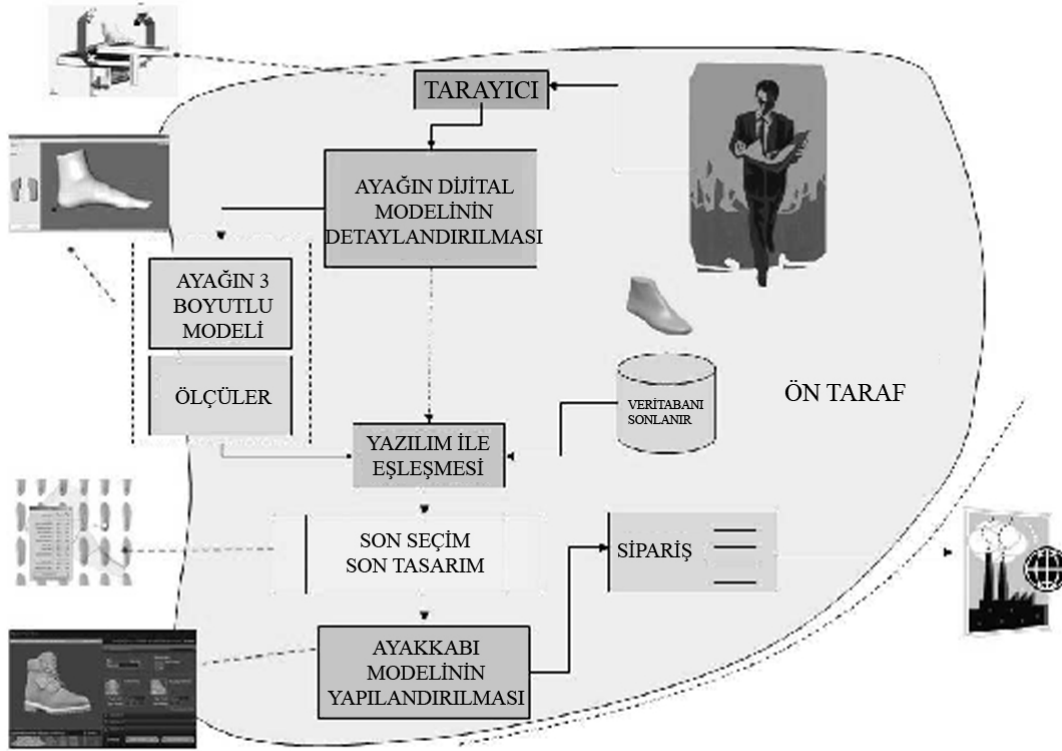


Şekil 2.2 : Goldenhook örgü firmasının web sitesinin, kullanıcı seçimleri ile ilerleyen arayüzleri (<http://www.goldenhook.fr>).

Kullanıcı ekranda gördüğü modellerden birisini, şapkanın boyutlarını, iplik cinsini, örgüde kullanılacak renkleri ve renk sıralarını ve en son olarak da istediği şapkayı örmesi için büyükannelelerinden birisini seçerek siparişini tamamlıyor (Şekil 2.2). Böylece kullanıcı internet bağlantısı olan dünyanın herhangi bir yerinden, sadece kendisi için özel olarak örülen bir bereye sahip olabiliyor (<http://www.goldenhook.fr>).

Ayakkabı firması Custom Foot'un, Westport'ta bulunan deposunda ayakkabı yoktur; buna rağmen 10 milyon çift farklı stil ve farklı büyüklükte ayakkabıyı üretebilme kapasiteleri vardır. Müşterilerinin farklı türlerdeki ayakkabı isteklerine cevap verebilmek ve farklı ayak büyüklüklerine göre üretebilmek için işbirliği ile bireyselleştirme yöntemiyle çalışan bir sistem kullanmaktadırlar. Mağazaya gelen müşterinin önce ayak uzunluğu ve genişliği en küçük farklılıkları algılayacak şekilde elektronik olarak taramaktadır ve bu bilgiler bilgisayara aktarılmaktadır. Daha sonra kullanıcı tarafından tercih edilmeyen ayakkabı modelleri program tarafından engellenerek müşterinin istediği özellikler ile eşleşen modeller bilgisayar tarafından görüntülenmektedir. Böylece kullanıcı ayak yapısına en uygun olan ve en beğendiği modeli firma yetkilisi ile beraber tespit ederek sipariş edebilmekte ve bu ürün iki hafta içerisinde İtalya'daki fabrikada üretilip müşteriye gönderilmektedir. Boer ve

Dulio'nun , 'Mass Customization and Footwear: Myth, Salvation or Reality?'adlı kitaplarında bu işleyiş diyagramı bir çok dijital farklı ayak tarama yazılımları ile ele alınmaktadır (Şekil 2.3).



Şekil 2.3 : Ayakkabı kişiselleştirme diyagramı (Boër & Dulio, 2007).

İşbirliği ile bireyselleştirme yönetiminin en belirgin şekli inşaat ve büyük proje çalışmalarında görülmektedir (Lampel & Mintzberg, 1996). Popülasyonun büyük çoğunluğu kişiselleştirmeyi, en çok yaşanan yapma çevreyi düzenlemek için istemektedir. Bu da yapı tasarımında, elemanlarında, bileşenlerinde, malzemelerinde ve hizmetlerinde kişiselleştirmeye diğer sektörlerle göre daha fazla yer verilmesini sağlamaktadır.

2.3.2 Uyarlanmış bireyselleştirme

Bu yöntemde, birbirinden farklı kullanıcıların değişik isteklerine göre cevap verebilen son ve standart bir ürün elde edilmektedir. Tek standart bir ürün sunulması kitlesel üretimi çağırırsa da ürünün tasarım aşamasında farklı kullanıcılar için olabilecek farklılıkların önceden düşünülmesi ve kullanım aşamasında müşterilere göre kişiselleştirilebilmesi vurgulanması gereken noktadır. Yükseklikleri ayarlanabilir bisiklet oturakları, ofis sandalyeleri, çizim masaları, yükseklik ayarlı otomobil koltukları ve açıları ayarlanabilir dikiz aynaları ilk akla gelen örneklerdendir. Her birey kendi ebatlarına en uygun yüksekliği ve açığı elde ederek konfor seviyesini sağlamaktadır. Benzer bir örnek olarak, her bilgisayar yazılımı

standart bir ürün olarak piyasaya sürülmektedir ve bu yazılımlar kullanıcıların kendi ihtiyaç ve isteklerine göre en verimli halde kullanılmaktadır. (Pine, 1993; Gilmore & Pine, 1997). Uyarlanmış bireyselleştirmede ürün üretici firma tarafından değil, satın alındıktan sonra kullanıcının kendisi tarafından kendi zevk ve ihtiyaçlarına karşılık verecek şekilde uyarlanmaktadır.

Dünya çapında yaygın bir biçimde kullanılan işletim sistemi Microsoft Windows'un, ileri sürümlerinde farklı görsel öğeler sunmaya başlamasıyla bireyselleştirme yöntemi de etkin bir şekilde kullanılmaya başlanmıştır. Birçok özellikleri bakımından dış dünyaya kapalı bir paket program özelliği gösterse de, Windows performans özelliklerinden güvenlik seçeneklerine, görsel unsurlardan erişebilirlik yöntemlerine kadar birçok kısımda kullanıcıyı özgür bırakarak bireyselleştirmenin önünü açmıştır. Kişisel bilgisayarında bu işletim sistemini kullanan bir kullanıcı kolaylıkla işletim sisteminin performans özelliklerini değiştirebilir, masaüstü ve kısayol tuşlarını istediği şekilde düzenleyebilir. Kullanım açısından bir farklılık yaratmasa da işletim sistemi üzerinde bireyselleştirmeye yönelik imkanlar sağlanması, kullanıcıların ürünü benimsemesine ve daha etkin bir biçimde kullanmasına katkıda bulunmaktadır. İşletim sisteminin kullanıcı tercihlerine göre yapılanmaya açık olmasının yanı sıra, üzerinde çalışacağı donanımın özelliklerine ve kapasitesine göre de özelleşmesi kitlesel bireyselleşmedeki esnekliğini göstermektedir.

Mimari uygulamalarda da, belirli bir yüklenici tarafından tamamlanan çok konutlu projelerde mutfak, banyo gibi hacimlerin çoğunlukla konutu satın alan kullanıcıların kendi zevk ve yaşam standartlarına uygun seçilmesi için müşterilere bırakıldığı bilinmektedir. Bu yolla kullanıcılar farklı teknoloji ve malzemelerle kendi yaşam alanlarını, projenin kendilerine sağladığı belirli ölçülerde kişiselleştirebilmektedir.

Marmol Radziner, hazır prefabrike elemanlar üreterek bunları konuta dönüştüren bir firmadır (Şekil 2.4). Bu firma, kullanıcılara farklı plan tipleri sunarak kendi yaşam standartlarına ve halihazırda bulunan arazilerine en uygun olan seçimi yaptırmaktadır. Daha sonra katalog sistemleri üzerinden kullanıcıya dış mekanda kullanılacak cephe kaplama, güneş panelleri, teras gibi yapı elemanı ve malzeme seçimi, iç mekanda kullanılacak döşeme, mutfak dolabı malzemesi, pencere gölgeleme sistemi, ıslak hacim zemin malzemesi, ıslak hacimde kullanılacak armatürler ve dolaplar gibi yapı elemanı ve malzeme önerileri sunulmaktadır (Şekil

2.5). Kullanıcı her aşamada, katalogda seçilen ürünlerin değişmesi ile fiyatın da ne kadar artıp azaldığını denetlemektedir. Bütün ürün seçimleri tamamlandıktan sonra firma tarafından seçilen ürünler ile üretilen prefabrike yapı, fabrikada imal edilerek araziye gönderilip montajı yapılmaktadır.



Şekil 2.4 : Marmol Radziner firması tarafından tasarlanan ev modeli görseli (www.marmolradzinerprefab.com).

Burada da sistem diğer uyarlanmış bireyselleştirmeler gibi daha önceden farklı kullanıcı senaryoları üzerine düşünülüp, üretim zincirine eklenmiş malzemelerden ve modüllerden oluşmaktadır. Firma farklı konut düzenlemelerini müşterilere bırakmaktadır. Her müşteri kendi isteklerine ve ihtiyaç programına göre kendi evini ve malzemelerini kendisi şekillendirebilmektedir. Kullanıcı katalogda sunulan bireyselleştirme ile sınırlı kalıp, katalog dışında herhangi bir yeni kişiselleştirmeye gidememektedir.



Şekil 2.5 : Marmol Radziner firması tarafından tasarlanan ev modelinde kullanıcının sistem ile etkileşimli katalog üzerinden malzeme seçimini gösteren arayüzü (www.marmolradzinerprefab.com).

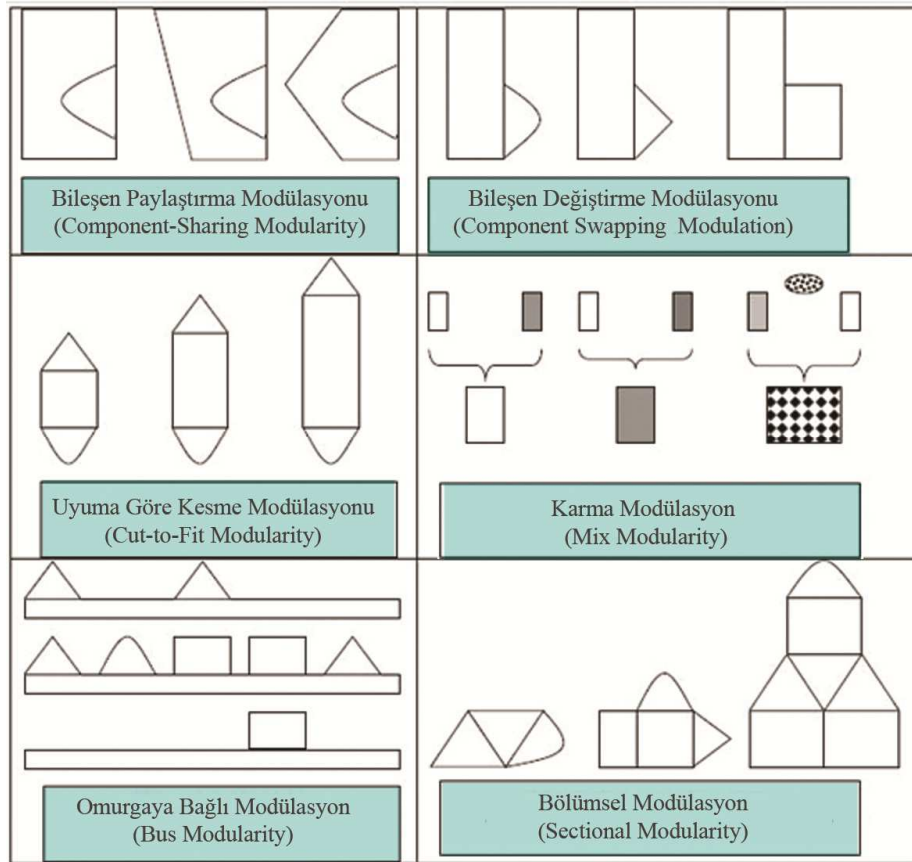
2.3.3 Modüler hale getirme

Kitlesel bireyselleştirmenin üretim fazında kullanılan yöntemlerin en başında gelen ve yoğun olarak kullanılan yöntem, modüler hale getirmedir. Bu yöntemde sonuç ürünü bir araya getiren bileşenler tüketici isteklerine göre farklı şekil ve çeşitlilikte birbirlerinin yerine kullanılarak, farklı özelliklerde ve varyasyonlarda ortaya çıkmaktadır. Modüler hale getirme yöntemiyle, kullanıcıların isteklerine göre satış sırasında arzu edilen farklı modüllerin bir araya getirileceği gibi, satış sonrasında da zaman içinde değişen isteklere uygun olarak bazı modüller değiştirilebilir.

Üretim yapan birçok şirket, aynı zamanda finansal olarak ayakta kalabilmek için pazarda mümkün olan en çok çeşitlilikte ürünü en az yer kaplayarak sağlama problemiyle karşılaşılıyor (Bi & Zhang, 2001). Bu sorunu aşmak için otomotiv, bilgisayar, elektronik, inşaat gibi sektörler modülasyon sistemini kullanıyorlar. Modüler hale getirme sistemi, karmaşık bir ürünü, daha küçük alt sistemlere bölüp, her bir bölümün bağımsız olarak tasarlanıp, üretilerek daha sonra bir araya gelen diğer modüllerin beraber bir bütünü oluşturmalarıdır. Modülasyon ile üretim aşamasında farklı üreticilerin farklı modülasyon sistemlerini tedarik edebilmesi, araba gibi çok parçadan oluşan bir ürünü karmaşık iletişim ve tedarik zinciri olmadan halledilebilir kılmıştır.

Örnek olarak; Mercedes Benz, Alabama’da bulunan üretim merkezinde, hava yastığı, direksiyon simidi gibi öğeleri içeren sürücü kokpitlerini ayrı bir modül olarak tasarlayıp, bunun sorumluluğunu da Delphi Automotive Systems’e vermiştir. Böylece farklı birçok üretici ile çalışmadan kaynaklanacak sorunlar tek bir firmanın üretimi gerçekleştirmesi ile aşılmıştır.

Åhlström ve Westbrook (1999), yaptıkları çalışma ile modüler hale getirme yöntemi ile kitlesel bireyselleştirmenin üretimdeki etkinliği arttırdığını ortaya koymuştur. Buna göre modüler üretim yaklaşımı, kişiye özel olarak kitlesel üretim yapan kuruluşların birçoğu tarafından kullanılmaktadır. Modüler hale getirme yöntemi üretim fazında olduğu kadar, aynı zamanda tasarım ve diğer aşamalarda da kullanılmaktadır. Burada da farklı modüler hale getirme türleri farklı aşamalarındaki örnekler ile açıklanacaktır. Ulrich ve Tung (1991), Pine (1999) çalışmalarında ürünlerde kullanılan farklı modüler üretim ve tasarım yaklaşımlarını, modüler hale getirmenin altı yöntemi adı altında incelemiştir. Bu modüler yaklaşımlar Şekil 2.6’da gösterilmektedir.



Şekil 2.6 : Ürün ve hizmetlerin kitlesel özel üretim yöntemi ile üretilebilmesi için altı modülasyon tipi (Ulrich ve Tung, 1991; pine, 1999).

2.3.3.1 Bileşen paylaşırma modülasyonu

Bileşen paylaşırma modülasyonu (Component-Sharing Modularity) ile, aynı bileşen parçasının veya aynı detayın birçok farklı üründe ortak olarak kullanılması sağlanmaktadır. Ortak kullanılan bu modül için her farklı tasarımda belirli bir hacim gerekmektedir. Bu yöntem ile maliyeti daha düşük ürünler elde edilmektedir ve üretimin çeşitleri arttırılmaktadır. Bu yöntem basit olarak, flash bellek giriş yuvasının her tür bilgisayarda aynı olması ve farklı dış yapıya sahip tüm flash belleklerin bu yuvalara uyum sağlaması örnek verilebilir.

Kas Oosterhuis'in tasarım ekibi olan ONL ile beraber tasarlayıp, 2006 yılında Hollanda Utrecht'te A2 yolunda inşaa edilen Akustik Duvar (Acoustic Barrier) projesi kütle olarak 1.5 kilometre uzunluğunda ve 8 metre yüksekliğindedir (Şekil 2.7). Akustik duvarın çelik strüktürü herbiri birbirinden farklı boyut ve şekillerde olan gri renkli, güçlendirilmiş 8300 adet 6mm kalınlığında üçgen cam ile kaplanmıştır. Strüktür, birçok çelik kirişin kafalarının aynı noktada birleşmesiyle oluşmaktadır. Değişmeyen çalışma prensibine sahip olan çelik taşıyıcıların düğüm noktası, birbirinden şekil ve boyut olarak ayrı olan bütün camların altında vardır ve bu bileşen detayı tüm sistemi ayakta tutmaktadır. Bileşen paylaşırma modülasyonuna göre aynı detaya sahip bu nokta tasarımı bütün bir mimari kütlede değişmeyen ve her taşıyıcı birleşiminde kullanılan ortak modüldür (Şekil 2.8).



Şekil 2.7 : Akustik duvar (<http://www.archdaily.com/15400/acoustic-barrier-onl>).

Geometrik şeklin mühendisliği ONL tarafından üstlenilmiştir. ONL ekibi, binlerce birbirine geçmiş elemanların verisinin (data) kusursuz olmasından tamamen sorumlu olmuştur. Bu amaçla, fabrikada üretim için kullanılacak tüm geometriyi tanımlayan bir betik (script) geliştirmişlerdir ve böylece yapıya ilişkin veriler bir veritabanında toplanmaktadır. Bu veriler üretim yapacak makinelere doğrudan okutulmuştur. Bu yolla mimarın 3 boyutlu modeli ile imalatçının üretim makinaları arasında doğrudan bir ilişki kurulması sağlanmıştır. Sonuçta mimarın üretilmesini istediği birbirinden farklı her parça kusursuz olarak imal edilebilmiştir (<http://www.archdaily.com/15400/acoustic-barrier-onl>).

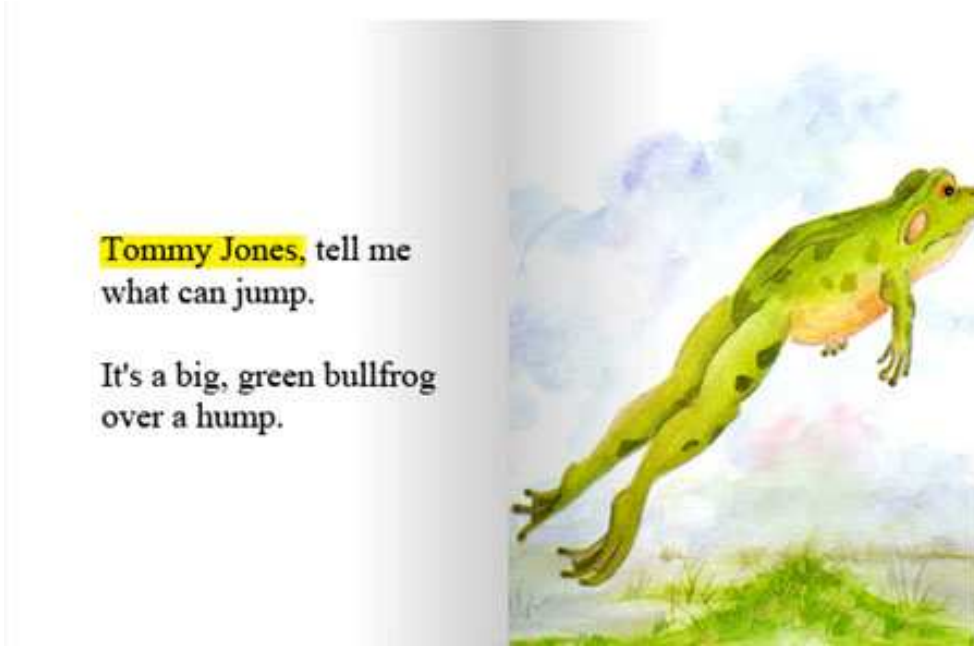


Şekil 2.8 : Akustik duvar detayları (<http://www.archdaily.com/15400/acoustic-barrier-onl>).

2.3.3.2 Bileşen değiştirme modülasyonu

Bileşen Değiştirme Modülasyonu (Component Swapping Modulation) ana üretim parçasının sabit kalarak, farklı tipte eklenen modüllerin ürünü tamamladığı ve farklılaştırdığı modülasyon tipidir. Çalışma anlayışı açısından bileşen paylaşma modülasyonunun tamamlayıcısıdır ve aynı zamanda birbiri içine geçmiş, birlikte düşünülmesi gereken iki kavramdır. Bu modülasyon tipinde değişimi sağlayacak farklı ürün bileşeni kadar çeşitlilikte ürün üretilebilir. Basit bir örnek olarak, artık dijital baskı yapan kopyalama merkezlerinde üretimin talebe göre yapılması verilebilir. Bu gibi yerlerde henüz üzerine herhangi bir işlem yapılmamış temel haldeki posta kartı, kitap, kahve kupası, yapboz, kitap ayracı gibi ürünler bulunmaktadır. Kullanıcılardan gelen istek üzerine kullanıcıların isimleri, seçtikleri fotolar bu ürünlere lazer baskı ile işlenmekte ve kullanıcı için sadece o kullanıcıya özel olarak farklılaştırılmaktadır.

Create-A-Book© çocuklar için özel kitaplar satan bir kitapçıdır. Standart olmayan bir sistem ile kitaplar raflarda değil, bilgisayar hafızalarında saklanmaktadır. Bir kitap satın almak isteyen müşteri, kitabı alacağı çocuğun özel bilgilerini firma ile paylaşır, firma bilgisayar aracılığı ile çocuğun bilgilerini müşterinin seçtiği bir standart hikayenin içerisine koyar ve kitap lazer baskıda basılıp ciltlenerek sadece bir çocuğa özel olarak üretilmiş şekilde müşteriye teslim edilir (Şekil 2.9). Burada sabit bileşenler kitabın ana akışı ve hikayesi iken, değişen bileşenler çocuğun adı, soyadı, yaşı, anne ve babasının adı, doğum yeri gibi kişisel bilgileridir. Dakikalık klavye hareketleri sonucu, sonsuz sayıda kitlesel bireyselleştirilmiş ürün elde edilmektedir (<http://www.createabook.com/>).

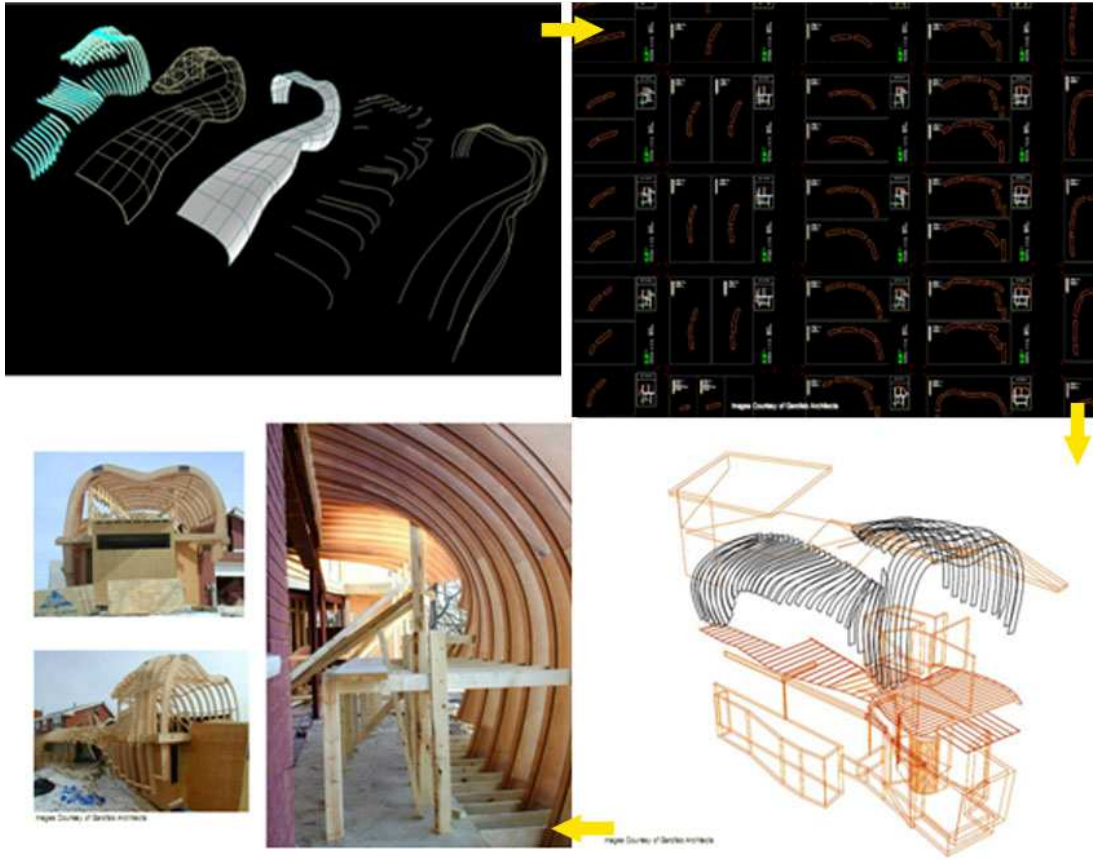


Şekil 2.9 : Create-A-Book© örnek sayfa (<http://www.createabook.com>).

2.3.3.3 Uyuma göre kesme modülasyonu

Bu tip modülasyon (Cut-to-Fit Modularity) bir bileşenin, başka bir bileşene takılmasından önce istenilen ebatlara göre şekillendirilmesine olanak tanıyan bir yaklaşımdır. Ürüne ilişkin birleşim detayları belirli sınırlar içerisinde aynı kalmak koşulu ile ebatların değişime uğraması yöntemidir. Burada kişiselleştirme, müşterinin istediği ebatların sağlanması ile elde edilmektedir. Çok yaygın olarak hazır giyim endüstrisinde bu yöntem kullanılmaktadır. Örnek olarak Custom Cut Technologies, satışa sunduğu takım elbiselerinde, müşterinin ölçülerine göre ceket boyu, omuz genişliği, kollar gibi değişkenleri uyuma göre kesme modülasyonu ile üretmektedir (Pine, 1993).

Manilow Evi projesi, ön tasarımdan inşaat aşamasına kadar tamamen dijital olarak adım adım üç boyutlu tasarım araçları kullanılarak tasarlanmış ve üretilmiştir. Form bulma arayışında Maya Animation Software destekli, bilgisayarda üretilen (computer-generated) tasarım formları ile çalışılmıştır. Bilgisayar destekli form arama çalışmasında strüktürün başlangıç ve son taşıyıcısı şekil olarak belirlenmiştir ve daha sonra ara taşıyıcı formlar üretilmiştir. Üretilen noktalar, çizgilere, bu çizgiler de kapalı yüzeylere dönüştürülmüştür.



Şekil 2.10 : Manilow evi, case study: Garofalo Architects, 2007.

Üretim aşamasına hemen geçmek yerine Mimar Douglas Garofolo, modelin bilgisayar üzerinde yaşamasına, günden güne gelişerek formun değişmesine izin vermiştir. Taşıyıcı strüktürü oluşturan omurganın her parçası ve birleşim detayları tasarımcı tarafından bilgisayarda üretilmiştir (Şekil 2.10). ‘Kaburga’ adını verdikleri taşıyıcılar mühendislik firması tarafından taşıyıcılığı test edildikten sonra, bilgisayar modeli CNC (Computer Numerical Control) kesimi yapacak marangoza web ortamında gönderilmiş ve strüktürü oluşturan bileşenler burada kesilmiştir. Kaburga bileşenler üç katlı lamine kontrplaktan oluşmaktadır ve yerinde birleştirilmek üzere

inşaat alanına gönderilmişlerdir (Case Study: Garofalo Architects, 2007). Burada bir bileşenin, diğer bileşene sabitlenirken oluşturduğu detaylar aynı kalmaktadır; uyuma göre kesme modülasyonu ile değişim gösteren öğeler, binanın formuna göre büyüyüp küçülen kaburgaların ebatlarıdır. Eğer form bulma çalışmalarında farklı müşterinin istediği senaryoya göre farklı sonuçlar üretilseydi, yine aynı birleşim detaylarına sahip farklı boyutlu kaburgaların kesilerek imal edilmesiyle oluşan, farklı hacimde bireyselleştirilmiş bir ev üretilmiş olacaktı.

2.3.3.4 Karma modülasyon

Karma Modülasyon (Mix Modularity) yönteminde, bileşenler kendi karakter özelliklerini yitirerek birbirleriyle kaynaşmakta ve ortaya birbirinden farklı ürünler çıkmaktadır. Sonuç ürün eklenen bileşenlerin özelliklerini yansıtsa da, içerisinde bileşenlerin tamamen ayırt edilmesi mümkün değildir (Pine, 2003). Bunun örneği, müşterinin istediği ama herhangi bir renk kartelasında bulunamayan duvar rengini, birkaç rengi karıştırıp elde eden boya firması ile verilebilir. Elde edilen boya ne tek tek bileşenlerden biridir, ne de sonuç boyanın içinden tek bir bileşen rengi elde edilebilir. İnternet üzerinden bitkisel ilaç ve enerji takviyesi satışları yapan bir firma olan Şekil 2.11’de görüldüğü üzere Youbar, müşterilerinin istekleri doğrultusunda kendi özel enerji takviyeli besinlerini hazırlamasına yardım etmektedir. Seçilen miktarda ve çeşitte protein, fıstık, tohum, kuruyemiş, tatlandırıcılar, kakaolar bağımsız olarak bir pakette karışım haline getirilmekte ve müşterinin karışımına verdiği isim paketin üzerine basılmaktadır. Bileşenlerin tek tek bulunması zor olmasa da istenilen ürünlerin, istenilen ölçülerde ve istenilen kıvamda karıştırılıp müşteriye özel olarak sunulması, müşterilerin kişisel tercihlerini elde ederken hayatını kolaylaştırmaktadır.



Şekil 2.11 : Youbar müşterilerinin istekleri doğrultusunda karışımlar hazırlamaktadırlar (<http://youbars.com>).

Bir ahşap firmasına iki kapaklı dolap yaptırmak isteyen bir kullanıcı, önce dolabın ebatlarına, sonra tercih ettiği malzemenin lamine veya masif ahşap olacağına, kaplama cinsine, kaplamanın rengine, kulpların cinsine veya kulpsuz istediğine, ayak veya baza seçimine, raf aralıklarına karar verir. Bu kararlar sonucu sadece o müşteriye özel bir üretim yapılır ve bu imalatla kullanılan bütün bileşenler, bir karma modülasyon sağlar. Bileşenler tek başlarına kullanılamayacağı gibi bir araya gelince bir bütünü oluşturmaktadırlar.

2.3.3.5 Omurgaya bağlı modülasyon

Omurgaya Bağlı Modülasyon (Bus Modularity) hâlihazırda bulunan bir temel üzerine bir veya birden çok tamamlanmış veya tamamlanmamış birimin eklenmesine olanak sağlayan bir sistemdir. Raylı aydınlatma sistemi (track lighting) omurgaya bağlı modülasyon için bir örnektir (Pine, 1993). Birbirinden farklı aydınlatma spotları aynı raya eklenerek, farklı dizilimler ile bir sergi mekânındaki aydınlatma sistemini eşsiz kılabilir. Bu tip modülasyonun bir örneğini otomotiv sektöründe de gözlemleyebiliriz. Bir Land Rover Jeep almak isteyen kullanıcı, standart sunulan omurgaya koltuk ısıtma, bluetooth ve navigasyon eklentisi, deri koltuklar, yan hava yastıkları, sabit hız kontrolörü gibi ilave birimlerin ekstra ücretlerini ödeyerek arabasını bireyselleştirebilir.

2.3.3.6 Bölümsel modülasyon

Bu modülasyon(Sectional Modularity) sistemi diğer modülasyon tiplerine oranla en fazla sayıda çeşitlenme ve kişiselleştirme olanağı sağlayan yöntemdir. Burada standart olarak sunulan ana bileşene, birden çok sayıda ve çeşitlilikte alt bileşen eklenebilmektedir. Eklenen her bileşenin diğer bileşenler ile birleştirilebilme kuralı, bu modülasyon tipini diğerlerinden ayırmaktadır. Eklenebilecek bileşenlerin miktarı müşterinin isteğine göre azaltılıp artırılabilir. LEGO oyuncakları bu tip modülasyona verilecek en açıklayıcı örneklerden birisidir. Standart bileşim özelliği olan bloklar, değişik tasarımlar ile bir araya getirilebilmektedir.

2.4 Seri Üretim ile Kitlesele Bireyselleştirmenin Kıyaslanması

Seri üretimden neredeyse yarım asır sonra ortaya atılan kitlesele bireyselleştirme fikri arasındaki üretim farklılıkları şöyle sıralanabilir:

Seri Üretimin Esasları:

- Standartlaşmış ürünler bir bant hattı üzerinde büyük miktarlarda üretilir.
- Aynı tip işi yapan birçok işçi çalışmaktadır.
- Genellikle çok sayıda, kısmi olarak homojen tüketiciye hitap eder.
- Seri üretim süreci sermaye yoğunluklu işler; çalışanlar işlerini ne kadar çok öğrenirlerse işi o kadar hızlı ilerletirler ve az zamanda daha çok ürün üretilir.
- Oluşabilecek kullanıcı hataları en aza indirilebilir, ürünlerin kalitesi artar.
- Ürünlerin bir defa üretildikten ve monte edildikten sonra değiştirilebilme özellikleri sınırlıdır.

Kitlesele Bireyselleştirmenin Esasları:

- Kitlesele bireyselleştirme kullanıcıya bir üründe veya hizmette istediği gerçek kişiselleştirmeye karar vermesine izin verir ve bu ürünü veya hizmeti sıradan bir seri üretim maliyetine elde etmesini sağlar.
- Ürünün her kullanıcı için kişiselleştirilebilmesine olanak sağlayan, esnek bilgisayar destekli üretim sistemlerini kullanır.

- Sipariş üzerine çalışma (build – to order) metodu sayesinde, bir istek gelene kadar herhangi bir ürün üzerinde çalışılmaz, böylece maliyet en aza indirgenebilir.
- Ürünler, bağımsız müşterilerin isteklerini tatmin etmek üzere yapılırlar.
- Yazılım tabanlı ürün yapılandırmaları sayesinde, ürünlerin ana bileşenlerinde ekle/çıkar yöntemi ile fonksiyonlar üretilebilir.
- Üretici ve tüketici arasında bağ kuran, ürünü kullanıcıya tanıtan ve kullanıcının gerçek isteklerini sorgulayan sistem yapılandırmaları veya interaktif ara yüzler, zamanla kitlesel bireyselleşmenin en önemli yapı taşı haline gelmektedir.

Kitlesel bireyselleştirme hızla seri üretimin yerini almaktadır. Seri üretim çok miktarda uygun maliyetli üretim sağlamayı amaçlamaktadır, fakat kitlesel bireyselleştirmedeki temel amaç maliyet değil, kültürel üretimdir. Başka bir deyişle, sanayi tarafından hazır üretilmiş ürünlerden birini seçmektense, müşteri tasarım sürecinin en başından üretimin sonuna kadar seçenekleri kendisi belirler (Kieran ve Timberlake, 2004).

2.5 Konut Tasarımında Kitlesel Bireyselleştirme

Dünyanın birçok yerinde düşük veya orta gelirli insanlar, ne tasarım ne de inşaat aşamasında mimarlardan veya mühendislerden profesyonel yardım almadan, geleneksel yöntemlerle kendi evlerini inşa ederler (Gomez, 1998). Bunun birçok sebebi vardır; birincisi ekonomik faktörlerdir. İnsanlar, inşaat sürecinin tamamında veya bir kısmında böyle bir profesyonel yardım almanın cesaret kırıcı düzeyde pahalı olduğunu düşünürler. Gerçekte de nüfusun büyük çoğunluğu bunu finansal olarak karşılayamaz. İkincisi, mimarlar daha çok yüksek gelirli müşterilerle çalışmayı veya geniş ölçekli projelerde yer almayı fakir insanlara konut yapmaya tercih ederler. Üçüncü ve belki de en önemli neden ise ‘insanların kendi evlerini nasıl yapacaklarını bilmeleridir’ (Gilbert,1994). Akıllarındaki bir fikir ile başlar ve belirli bir zaman ve büyük çabalar sonucunda kendi gerçek konut ihtiyaçlarına cevap veren evi yaparlar (Noguchi & Hernandez , 2005).

Geçen on yıllarda, sabit gelirli insanların konut aldığı kooperatif projelerinde, bir kurumda çalışanların barınması için yapılan toplu konutlarda veya az geliri olan

insanlar için tasarlanan sosyal konutlarda genellikle tip projeler inşa edilmiştir. Günümüzdeki bu tür projelerde, potansiyel olarak evi kullanacak olanlar ne tasarım sürecinde ne de inşaat aşamasında proje hakkında söz sahibi değildirler (Gomez, 1998). Bu tür toplu konutlarda, hazır, standardize edilmiş mimari projeler kullanılmaktadır. İnsanların kendi kullanacakları evleri tasarlayabilme ve inşa edebilme yeteneklerinin olmasına karşın, ev alıcıları tamamen bu tasarım sürecinin dışında tutulmaktadırlar. Bu da demektir ki, gerçek konut ihtiyaçları ile üretilen konutların şartları çoğu kez birbirlerini karşılamıyor ve insanlar yeni ‘yaşam alanlarına’ geçtikten sonra konutlarında değişiklik yapma ihtiyacı hissediyorlar (Noguchi & Hernandez , 2005).

İnsanlar hayat standardı olarak daha konforlu bir şekilde yaşamayı her zaman tercih ederler, bu nedenle çoğunlukla bir ev edindikten sonra hemen yeni konutlarının kendilerine göre yaşam kalitesini yükseltmeye çalışırlar, yani kişiselleştirirler. Genellikle kullanıcının istek ve ihtiyaçları doğrultusunda, konutu yeniden düzenlemek amacıyla eve yeni elemanlar eklenir (Tippie, 2000). Evi kişiselleştirmeye duyulan istek her zaman vardır. Birçok toplu konut, orijinal halinden birimlerin ebat ve tarz olarak kullanıcıları tarafından farklılaştırılıp, bireyselleştirilmiştir. İnsanlar kendi konutlarını, komşularınınkinden farklı olacak şekilde yenilemektedirler, bu yenilik kimi zaman bir iç yapı elemanının değişikliği veya bir odanın eklenmesi veya çıkarılması olabilmektedir (Noguchi & Hernandez , 2005).

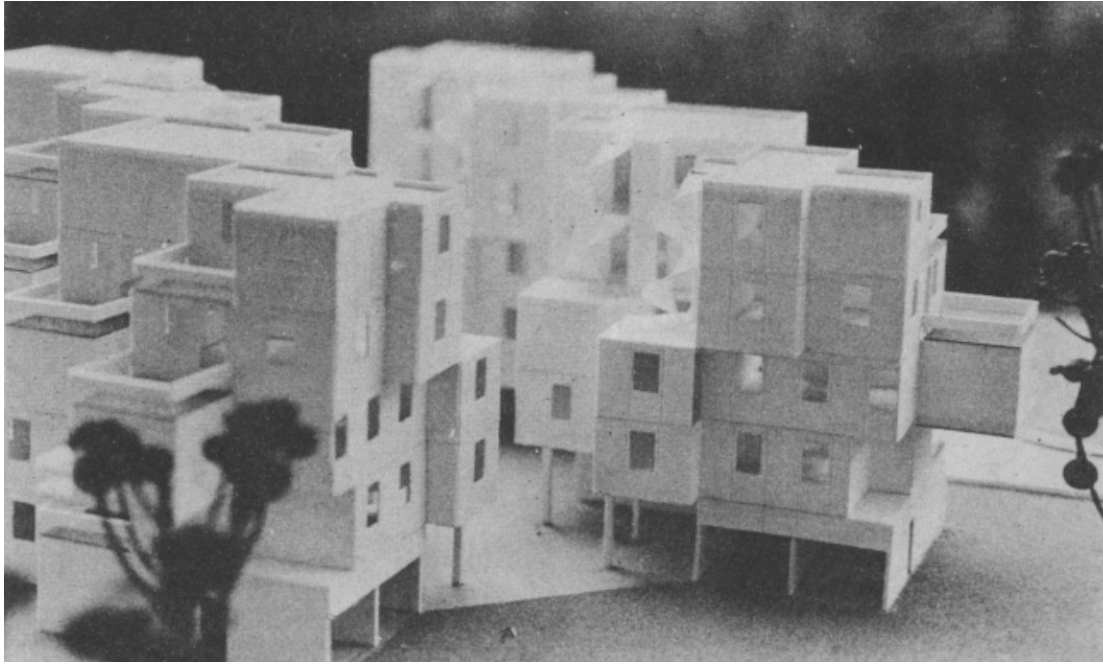
2.5.1 Konut projelerinde üretim yaklaşımları

Smith’e göre (Smith, 1998) günümüzde mevcut konut üretim yaklaşımları üç grupta toplanabilir; “kitlesel üretim bazlı”, “yarı bireyselleştirilmiş” ve “bireyselleştirilmiş”.

Kitlesel üretim bazlı yaklaşım ile çalışan inşaat yüklenicileri geniş ölçekli üretim yapmaktadırlar. Farklı müşteri profillerine uygun, genel piyasa araştırmaları sonucunda çoklu kullanıcı memnuniyetini sağlamak için birkaç ev modeli geliştirirler ve üretirler. Bu tip evlerin inşaat sürecinde yapacakları işleri iyice öğrenen ustalar, birbirinin hemen hemen aynı evlerde tekrarlayan işleri hızla yaparlar, benzer malzeme kullanmanın da avantajı ile zaman ve maliyet kaybı olmadan inşaat süreci hızla ilerler (Noguchi & Hernandez, 2005). Bu sisteme örnek olarak toplu konut projeleri verilebilir.

Yarı bireyselleştirilmiş tasarım yaklaşımı, kitlesel üretim bazlı yaklaşım ve bireyselleştirilmiş konut üretimi yaklaşımının arasında yer almaktadır. Burada genellikle kitlesel üretim bazlı yaklaşımdaki gibi önceden var olan planlar üzerinden veya hali hazırda var olan konut modellerine, müşterinin istekleri doğrultusunda iç mekân şekillenmesi, malzeme seçimleri, dekoratif öğelerin belirlenmesi gibi seçenekler sunulmaktadır. Bir planı revize etmek, yeni bir tasarım yapıp inşa etmekten daha hızlıdır ve daha az maliyetlidir (Smith, 1998).

Yarı bireyselleştirilmiş tasarım yaklaşımına örnek olarak 1971-1977 yılları arasında İzmit'te 'Birleşmiş Mimarlar' tarafından gerçekleştirilen "Yeni Yerleşmeler Projesi"si verilebilir (Şekil 1.2). Kullanıcının planlama kararlarına katılımını politik bir ilke olarak kabul eden proje, çoğu imalat sanayinde çalışan 30.000 dar gelirli işçi ailesine, katılımsal sürecin gereklerine tam olarak uyabilecek özel prefabrikasyon teknolojisi ile konut sağlamayı amaçlamaktadır (Çavdar, 1978).



Şekil 2.12 : Katılımcıların belirlediği konutların tasarımlarının maketi, İzmit yeni yerleşmeler projesi (Çavdar, 1978).

Projedeki ana fikir olan kullanıcıların kendi konutlarını tasarlayabilme imkanı için, projede Habraken teorilerini temel alan, yatay yükleri taşıyan yerinde dökülmüş bir çekirdek ve bunun etrafına monte edilen döşeme ve kolonlardan oluşan bir sistem uygulanmıştır (Şekil 2.13). Sistemin verdiği rahatlıkla, istenildiği gibi planlarla oynanabilmektedir ve modüler ızgara içinde değişen biçim ve büyüklük

farklılıklarını yansıtan binlerce farklı çözümler üretilebilmektedir (Bulca, 1977). İlk yıl üretimi için, kullanıcı olacak 1500 adet aile ile ön tasarım aşamasında anketler ve sohbetler düzenlenerek konut planları belirlenmiştir ve buna göre konutlar üretilmiştir. Bu süreçte izlenen aşamalar şöyledir:

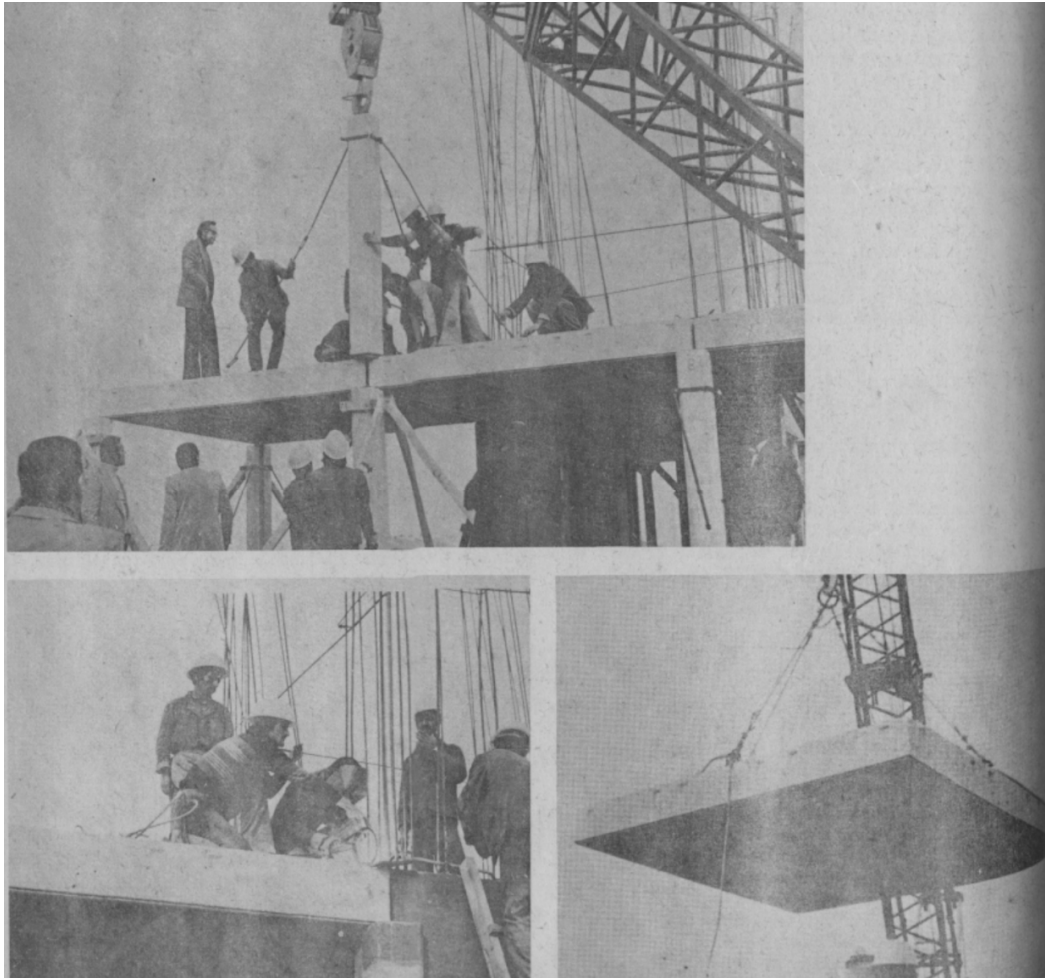
Aşama 1: Karar belirleyici konuşmanın tutanağı mekânsal gereksinimler formuna aktarılır.

Aşama 2: Yapı tercihleri kontrol edilir

Aşama 3: Yaşam tarzı tercihlerine göre eylem alanlarına verilen öncelik ve bu alanlar arasındaki ilişkilere bağlı olarak temel plan iskeleti tanımlanır.

Aşama 4: Tek tek ailelerin gereksinimlerine göre mekansal bileşenlerin sayıları ve yerleştirme şekillerine bağlı olarak seçilecek iki eylem alanı belirlenir.

Aşama 5: Seçilen iki eylem alanı temel plan iskeleti üzerinde birleştirilir.



Şekil 2.13 : Yerlerine monte edilen döşeme ve kolonlar, İzmit yeni yerleşimler projesi (Bulca, 1977).

Seilen eylem alanlarında kullanılan prefabrike elemanların sayısı ve gerekli malzeme miktarları nceden belirlendiėi iin, bu sre sonunda ortaya ıkan plan zmnn getireceėi toplam maliyet kendiliėinden belirlenebilmektedir. Bu toplamın, katılımcının (kullanıcı aile) satın alma gcne uymadıėı durumlarda planı (mimar), konuşmalarda saptanan tercih nceliklerine gre yeni eylem alanı zmlerini temel plan iskeleti zerine yerleřtirerek, planı mali olanaklarla uyumlu byklėe getirir (avdar, 1978). Sonunda katılımcının gerek isteklerine cevap veren, ekonomik olarak uygun konutlar inřa edilmiřtir.

Bireyselleřtirilmiř yaklařımda ise tasarımcı eřsiz bir evi tasarlamak iin temiz bir sayfa veya boř bir bilgisayar ekranı ile iře bařlar. Bireyselleřtirilmiř tasarım yaklařımı, yeni bir evi bireysel konut gereksinimlerine tamamen karřılık verecek řekilde projelendirmek demektir. Bununla beraber kiřiselleřtirme dzeyi arttıka standardizasyon dzeyi dřer (izelge 2.1). Bu srete tm veya bazı bileřenler standart olmayan yapı elemanlarından oluřacaėı iin maliyet artabilir veya inřaat sreci uzayabilir (Noguchi & Hernandez, 2005). İlerleyen biliřim teknolojileri (CAD vs) ile retimde kullanılan kesme – birleřtirme makinelerinin (CNC) eřėdml alıřması sonucu kitlesel bireyselleřtirmede konut retim sreci de kısaltılabilir ve daha ekonomik retim yolları elde edilebilir.

izelge 2.1 : Konutta standardizasyon ve bireyselleřtirme dzeyleri (Noguchi & Hernandez, 2005).

	Standardizasyon Dzeyi	Bireyselleřtirme Dzeyi
Kitlesel retim Bazlı Konut	Yksek	Dřk
Yarı Bireyselleřtirilmiř Konut	Orta	Orta
Bireyselleřtirilmiř Konut	Dřk	Yksek

Noguchi & Hernandez’e gre, konut mimarisinin kiřiselleřtirmeye uygun olması iin gerekenler řu řekilde belirtilmiřtir:

Bir konutu tasarlamak, inřa etmek ve pazarlamak hem rnleri ve hem de hizmetleri gz nne almayı gerektirir. Bir ev “rnler” olarak nitelendirilen birok bileřenden

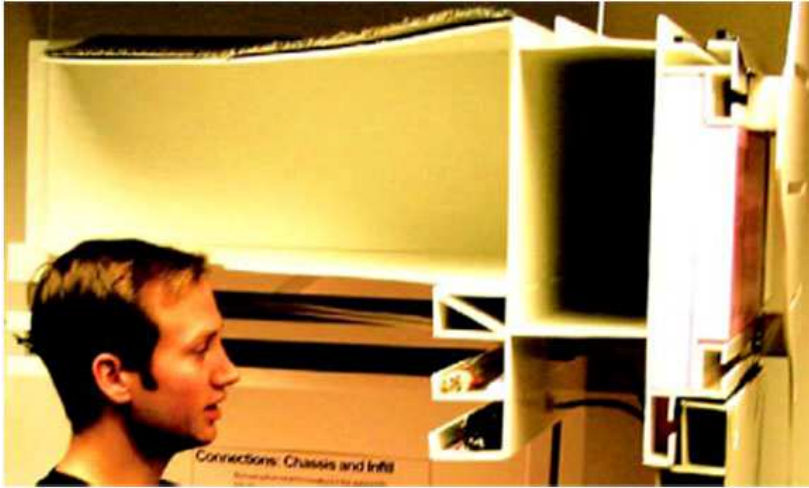
oluşurken, tasarım, inşaat ve pazarlama süreçleri ise “hizmetler” olarak tanımlanabilir. Bir toplu konut tasarımı için de bu iki kavram kullanılabilir, yapı malzemeleri ve sistemleri ürünlere, konut tasarımı ve inşası ise hizmetlere karşılık gelmektedir. Bir sistem bir ürünün tasarımı, üretimi ve satışı olarak görüldüğü zaman, kitlesel bireyselleştirme ürünlerinin ve iletişim hizmetlerinin kişiselleştirilmemesi düşünülemez. Bu kitlesel bireyselleştirme anlayışına göre, ürün alt sistemleri olarak konut bileşenlerinin seri üretilmesi ve tasarım hizmetlerinin son ürününün kişiselleştirmeye olanak tanıyacak şekilde kullanıcı ile etkileşimli hale getirilmesi gereklidir. Kişiselleştirilen ürünlerde, “kullanıcı katılımı” değişmez kural olduğundan, konut imalatçıları müşterilerine, seri üretilmiş standart bileşenler arasından seçim yaptırmak ve bir son ürün elde edilmesine olanak vermek amacıyla gerekli destek iletişim sistemlerini ve araçlarını sunmalıdır (Noguchi & Hernandez, 2005).

Noguchi ve Hernandez’in tartışmaya açtıkları bu bireyselleştirme yaklaşımları tasarım aşamasından başlayarak, üretim sürecinin sonuna kadar devam eden ve sadece bir proje için kullanılan bir kişiselleştirmeyi değil, seri üretilmiş parçaların farklı kombinasyonlarla inşa sürecinde kullanılarak elde edilen bir kişiselleştirmeyi tanımlamaktadır.

Smith’in 1998’de ortaya attığı ve Noguchi & Hernandez’in geliştirdiği, üç farklı tasarım ve üretim yaklaşımının dışında K. Larson önderliğinde, Massachusetts Teknoloji Enstitüsündeki (MIT) House’n grubu tarafından paralel şekilde konutta kitlesel bireyselleştirme üzerine iki farklı yaklaşım geliştirilmiştir. Birincisi, neredeyse herkesin tercihi olan, kurumsal yükleniciler tarafından üretilen, standart bir evi satın almak; ikincisi ise küçük bir azınlık tarafından tercih edilen süreç, maliyet ve oluşacak riskleri göze alarak bir mimarla çalışmak (Larson, 2004). Bu farklı iki ev üretim yapısını değiştirmek üzerine House’n grubu, açık kaynaklı yapı (open source building) konseptini ve bu kavramı destekleyen yeni inşaat metotlarını araştırıp, prototipler geliştirip test ederek yeni tasarım yolları arayan bir öneri geliştirmiştir. Bu önerilen yeni anlayışta bir mimarın tasarladığı bir konutun kalitesi ve duyarlılığını, yükleniciler tarafından birden fazla toplu üretilen evde kullanarak verimliliği en üst düzeye çekmenin olası olduğu belirtilmiştir.

Aynı tip standart evlerin yerini almasını bekledikleri açık kaynaklı yapı modeli yaklaşımında;

- Konut yüklenicileri, bireylere özel çözümler sunan anlaşmacılara dönüşmekte,
- Mimarlar, binlerce benzersiz tasarım yapan tasarım-makinaları tasarlamakta,
- Üreticiler oluşturulacak ürünlerin standartlaşmasını sağlamakta ve üretilen bileşenlerin tedarikçilerine dönüşmekte,
- İnşaatçılar, uygulamacı ve monte edici durumuna dönüşmekte,
- Müşteriler (konut alıcıları), karar verme aşamasında tasarım, ürünler ve hizmetler konusunda kişiselleştirilmiş bilgileri veren tasarımcılara dönüşmektedir (Larson, 2004).



Şekil 2.14 : Açık kaynaklı yapı (Larson, 2004).

House'n grubu, açık kaynaklı yapı yaklaşımında, yapının iki farklı ana bileşenden oluştuğunu düşünmektedirler. İskelet (chassis) yapıdaki standart bileşenleri; dolgu (infill) ise bağımsızca kişiselleştirilmiş ve iskelete standart yollarla bağlanan modülleri tanımlar (Şekil 2.14).

Açık plan yapısına sahip binada iskelet olan ana birim, binanın temel servislerdir ve dolgu olan kısım kullanıcı tarafından kişiselleştirilebilen, sipariş üzerine üretimi tamamlanan ve binanın iskeletine hızlıca monte edilebilen parçalarıdır (Larson, 2004).

Açık kaynaklı bina yaklaşımında uzman olmayan tasarımcılar (evi alan müşteri), toplu üretilen yapıda kendi konutlarını tasarım sürecinde seçimler aracılığıyla kişiselleştirme olanağına sahiptir. Bu tasarım kavramında uzman olan mimarlar kendi bilgi ve becerilerini kullanarak, uzman olmayan tasarımcılara karmaşık ve zor olan karar alma sürecinde yardımcı olacak sistemler ve programlar tasarlamaktadır (Larson, 2004). House'n grubunun sunduğu, açık kaynaklı bina kavramında kişiselleştirme binanın bazı birimlerinde kısıtlı olarak uygulanır; kişilere benzersiz konut tasarımı olanağı değil, belirli bir sistem içerisinde seçim ve uygulama şansı tanınır.

2.5.2 Kitlese bireyselleştirme yaklaşımıyla konut tasarımında bilişim teknolojilerinin rolü

Bilgisayar destekli mimari tasarımlar, müşterilerin bir üründe kendi istediklerini seçerken tasarıma, değişikliklere, analize veya tasarımın optimize edilmesine olanak verdiği için önemlidir. CAD sistemlerinin tasarım sürecinde kullanılması, müşterinin isteğinin iletilmesi ile ürünün alınması arasında geçen zamanı kısaltması, detaylarda hata payı çok düşük malzemenin elde edilmesi ve maliyet tahmininin yapılabilmesi ve dosyaların ve tasarımların standartlaştırılabilmesi açısından faydalıdır. Alıcının yeni evi olarak tasarlanan farklı modellerin, müşterinin seçimine sunulurken görselleştirilebilmesi olanaklıdır. Etkileşimli bilgisayar destekli tasarım sistemleri esasen iki boyutlu çizimlerin tasarımlarına hizmet etmesinin yanında, renkleri ve gölgeleri olan üç boyutlu katı modellemeler gibi gelişmiş geometrik modellemeler, mimara ve alıcıya grafik ekranından daha fazla bilgi vermesi açısından daha faydalıdır (Noguchi & Hernandez, 2005).

Yapıda birbirinden farklı olan bileşenleri standart parçalar gibi seri üretme olanağı, kitlesel bireyselleştirme fikrini bina tasarımı ve üretim alanına taşımıştır. Yapı elemanlarının tek tek özel olarak CAD araçları ile tasarlanıp, CNC makinelerinde üretimi, dijital mimaride artık mümkün olmaktadır. Bir CNC kesim makinesi için birbirinden farklı 1000 adet parçanın üretilmesi, birbirinin aynısı olan 1000 adet parçanın üretilmesi ile aynı zorluğa ve üretim maliyetine sahiptir (Kolarevic, 2003).

Bütün teknolojik gelişmelerin getirdiği tasarım ve dijital üretim imkanları, mimarları 2. Dünya Savaşı sonrasında gelişen seri üretim konut anlayışından (mass production), kitlesel bireyselleştirme (mass customizatin) ile üretilen konut tasarımına yaklaştırmıştır. Bilgisayar destekli tasarım ile dijital olarak komut verilen ve kontrol edilen kesme makineleri sayesinde, müşteri istek veya yaşama biçimi odaklı, tasarım çeşitliliğinin fazla olduğu ekonomik olarak maliyeti çok yüksek olmayan konutlar üretilebilmektedir. Bir bina içerisindeki aynı tip evlerin, her kesimden ve farklı hayat tarzına sahip insanlar tarafından kullanılmasının her müşterinin isteğini aynı derecede tatmin etmesini beklemek yerine, müşterinin bireysel olarak isteklerinin göz önünde bulundurularak, tasarım veya üretim aşamasında farklılaşmaya gidilmesi üretim olanakları ve üretim maliyetleri açısından olanaksız değildir.

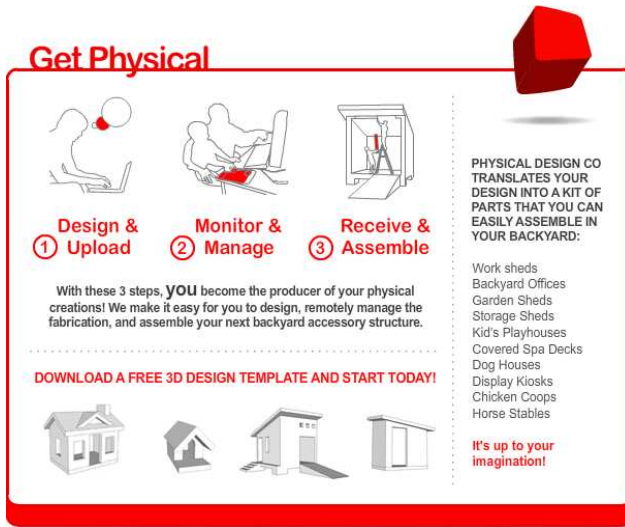
Kitlesel bireyselleştirme ile üretilen tasarımlar, konut bileşenlerinin standardize edilmesiyle faaliyet alanı ekonomisi elde edilerek üretim maliyetini düşürme potansiyeline de sahiptir ve konut alıcılarının yeni evlerini tamamıyla kendi istekleri doğrultusunda kişiselleştirebilme olanağı sağlar (Noguchi & Hernandez, 2005).

3. KONUT TASARIMINDA KİTLESEL BİREYSELLEŞTİRME ÖRNEKLERİ

Bu bölümde, kitlesel bireyselleştirme kullanılarak üretilen konut örnekleri sunulacaktır.

3.1 Get Physical

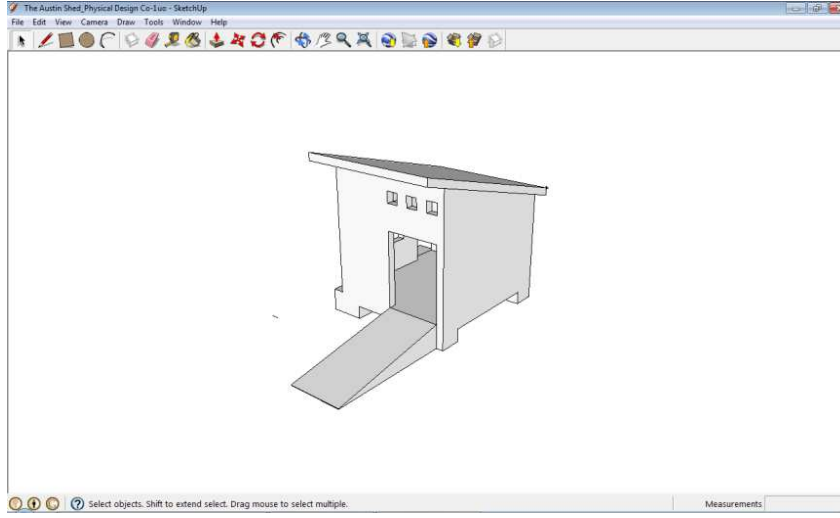
Physical Design Co. adlı şirketin internet üzerinden işleyen çalışma sistemi ile kullanıcılar, satın almak istedikleri basit yapıları kişiselleştirerek, üretime gönderip, malzemelerini kendileri bir araya getirip, kolayca istedikleri birimi elde edebiliyorlar (<http://www.physicaldesignco.com>). Web üzerinden işleyen sistemde kullanıcı ilk önce çalışma kulübesi, arka bahçe ofisi, teras, depo kulübesi, çocuk oyun evi, alet odası, köpek kulübesi, büfe vs. gibi birçok seçenek arasından hangi yapıyı istediğini belirler (Şekil 3.1).



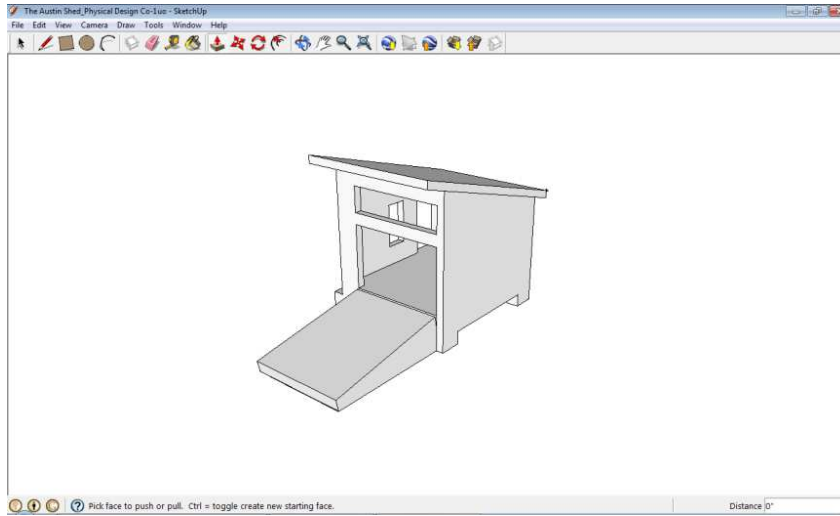
Şekil 3.1 : Get Physical arayüzü ekranı (<http://www.physicaldesignco.com>).

Ardından ücretsiz olan Google CAD yazılımı Sketch-up ile oluşturulmuş üç boyutlu bina çizimini bilgisayarına indirerek basit komutlarla istediği değişiklikleri yapar (Şekil 3.2; Şekil 3.3), tercih ettiği malzemeyi belirler, kendisine en yakın olan CNC

üretimi yapacak olan fabrikayı belirler ve maliyet hesabını yaparak tasarım aşamasını tamamlar.

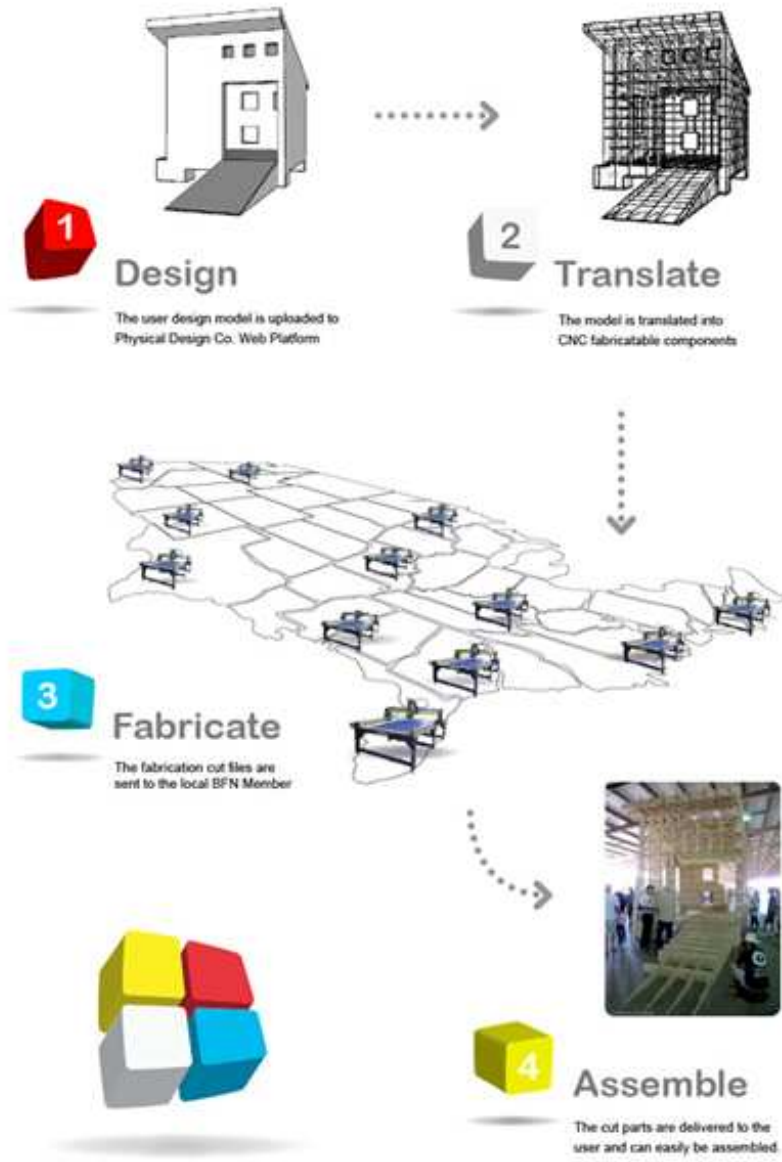


Şekil 3.2 : Google Sketch-Up programı ile oluşturulmuş hazır model (<http://www.physicaldesignco.com>).



Şekil 3.3 : Google Sketch-Up programı ile oluşturulmuş modeller kullanıcı tarafından kişiselleştirilmektedir (<http://www.physicaldesignco.com>).

Üretim aşamasında kullanıcı tarafından tercih edilen malzeme yine kullanıcıya en yakın olan fabrikada, üreticinin kararlaştırdığı malzeme birleşim detaylarına ve taşıyıcı esasına göre hazırlanan üç boyutlu puzzle mantığında CNC kesim yapılır, malzeme dişler aracılığı ve basit vidalama sistemi ile çalışmaktadır. Kesimleri yapılan parçalar, montajı yapılacak araziye gönderilir ve kullanıcı bileşenleri birleştirerek kendi yapı birimini inşa eder (Şekil 3.4; Şekil 3.5).



Şekil 3.4 : Birimlerin tasarım, dönüşüm, üretim ve birleştirme evreleri (www.physicaldesignco.com).



Şekil 3.5 : Malzemelerin üç boyutlu birleşim detayları (www.physicaldesignco.com).

3.2 Dry-In House

Dry-In House projesi, 2005’ deki Katrina fırtınasından sonra, New Orleans’da konutlarını kaybeden insanlar için, tekrar eski konutlarının yerlerine kısa sürede inşa edilebilecek, kullanıcıların ihtiyaç duyduğu altyapı sistemlerini (barınma, su, elektrik) sağlayacak ve aynı zamanda afetten etkilenen ailelere kendi evlerini kişiselleştirebilme imkanı sağlayan bir sistemdir. Bu sistemin FEMA’nın (Federal Emergency Management Agency) diğer doğal afet konutlarından farkı, geçici değil, kalıcı olarak kullanılabilmesi; belli bir yerde kurulan toplu afet konutlarından değil, her ailenin eski evlerinin yerlerine kuruluyor olması; esnek olmayan ve her yerde aynı ebatlarla çözülen (one fits all solution) seri üretim konseptinde değil, kitlesel bireyselleştirme olanağı sunarak seri üretimin monotonluğundan uzak olması; kullanıcılarına evlerini belirli bir zaman sonrasında, dış mekanın bitiş elemanları ve iç mekana eklenecek bölme elemanları ile evi geliştirebilme imkanı sunmasıdır. Aynı zamanda maliyet olarak her ikisi de 60.000\$ dır (Şekil 3.6) (Hecker, 2006).



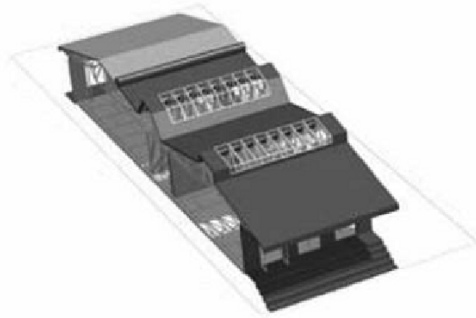
FEMA

Maliyet: 60.000\$

Genel

Tek Kullanımlık

Seri Üretilmiş



dry-in house

Maliyet: 60.000\$

Kişiyeye Özel

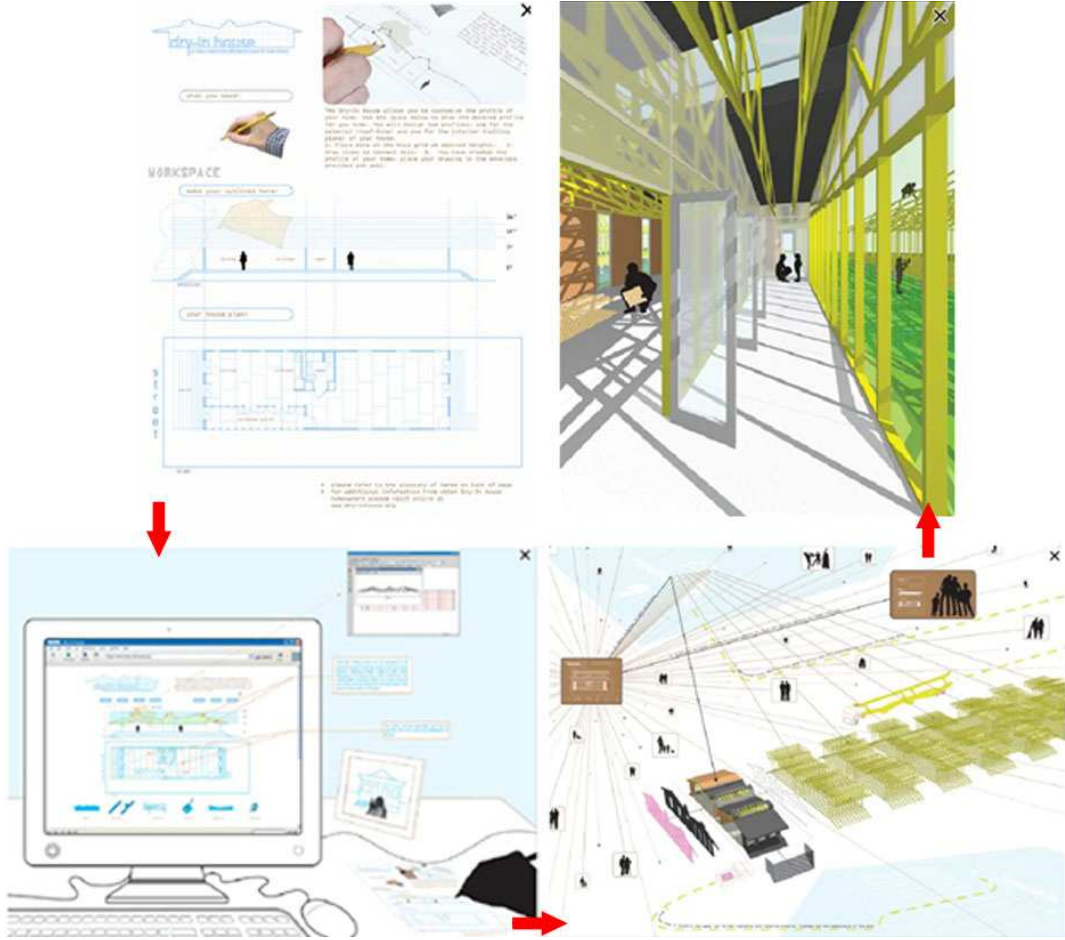
Kalıcı

Kitlesel Bireyselleştirilmiş

Şekil 3.6 : Afet konutları ile Dry-In projesinin karşılaştırılması (<http://ddbnola.org>).

Tasarım aşamasında ilk önce, gelecekte evin sahibi olacak olan kullanıcı, Dry-In House web sitesinde bulunan çalışma tablosuna erişir. Bu tabloya, kullanıcı istediği, kendi evini eşsiz kılacak şekilde kesitte çatı silüetini çizer, tavan camlarını yerleştirir, yatak odalarının sayısını işaretler. Sonra kullanıcı eskizini siteye yollar, tasarım ekibi evin yapı esaslarını geliştirir, makas taşıyıcıları ve kullanılacak panel sayısını vb.

belirlirler ve daha sonra üretimi yapacak ve malzemeleri gönderecek olan imalatçılara bu çalışmayı yollar (Şekil 3.7).



Şekil 3.7 : Dry-In ev projesi tasarım aşamaları (<http://ddbnola.org>).

Fabrikaya gönderilen dosyanın statüğü, özel durumlar için de lokal kodlar üreten parametrik modelleme yazılımı tarafından incelendikten sonra makas sistemin bileşenlerini sadece o proje için özel olarak üretecek program kodu elde edilir ve makas sistem bileşenlerinin kesimleri yapılır. Ev sahibinin tasarımına göre özel üretilen bu elemanlar, inşa sahasına geldiği zaman hızlıca yerlerine yerleştirilebilirler (Hecker, 2006).

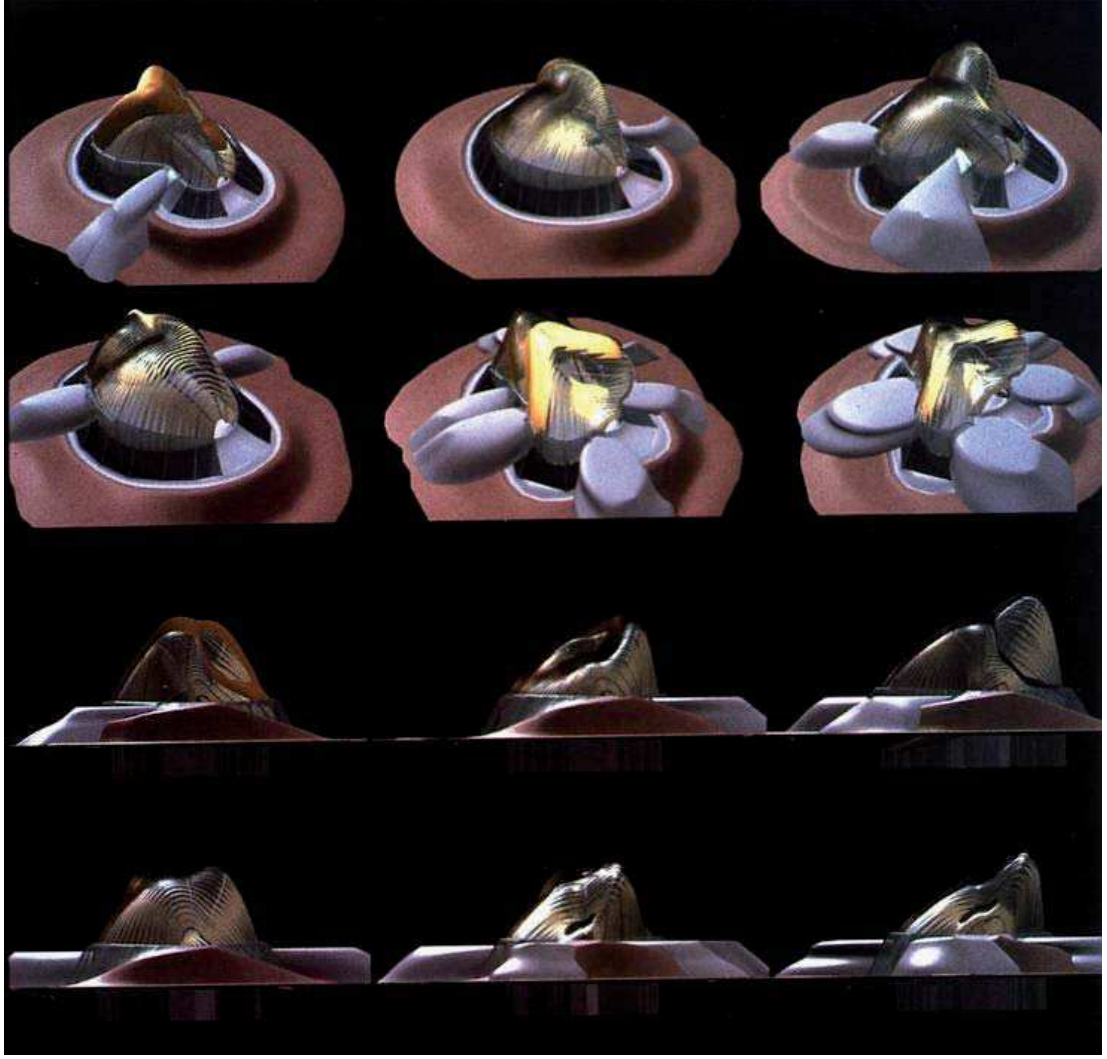
Dry-In konut projesinde, inşaat süreci üç ana bölümden oluşmaktadır, çekirdek, ana strüktür ve kabuk. Çekirdek olan kısım mutfak ve banyoyu kapsayan ve su tesisatı, elektrik ve iklimlendirmenin de içinde bulunduğu prefabrik birimdir. Ana strüktür binanın taşıyıcısı olan temeli ve kullanıcının tasarımı sonucu üretilen prefabrike makas sistemi (Şekil 3.8), kabuk ise prefabrike ve yalıtımlı zemin, çatı ve duvar panellerini içerir (<http://www.physicaldesignco.com>).



Şekil 3.8 : Dry-In ev projesi inşa edilen prototipi (<http://ddbnola.org>).

3.3 Embriyolojik Ev

Greg Lynn'ın 1997-2001 yılları arasında bilgisayar destekli tasarım ve genetik mutasyonun karışımından oluşan Embriyolojik ev projesi mimarlığın ve kitlesel bireyselleşmenin kesiştiği önemli örneklerden birisidir. Bu projedeki konut tasarımlarının kitlesel bireyselleştirilerek bağımsızlaşması, lineer olmayan dinamik süreçte parametrik çeşitliliğin başkalaşması ile sağlanmaktadır (Kolarevic, 2003). Embriyolojik ev ile konut anlayışı, modüllere dayalı üretilmiş dört köşeli modernist ev formu fikrinden, temel veya primitif formdan potansiyel olarak sınırsız türeyen evrimsel konut anlayışına doğru yönelmiştir. Greg Lynn'a göre; "İdeal veya orijinal Embriyolojik Ev yoktur. Herkes mükemmel evini mutasyonlar sonucunda elde eder" (Lynn, 2000).



Şekil 3.9 : Embriyolojik ev projesinin topoğrafya ile ilişkisi (Kolarevic, 2003).

Adaptasyon yeteneği yüksek olan ve kabarcık (blob) olarak tanımlanan bu strüktürler, her türlü hava şartları ve topografyaya uyum sağlayabilecek şekilde tasarlanmıştır (Şekil 3.9). Araziye bir nevi gömülerek oturan bu konutlar iki katlıdır ve otomotiv sanayi ürünleri olan arabaların taşıyıcı mantığı ile çalışan malzeme sistemi esnek, eğrisel ana çerçeve çelikten, ara kayıtlar alüminyum strüktürlerden ve bunları kaplayan kabuktan meydana gelmektedir. Bütün bu elemanlar dış yüzeydeki eğrisel hareketlerin yarattığı gerilimi dengeleyecek şekilde deniz kabuğu şeklini alarak belirli bir düzende çalışmaktadır. Lynn, bu projenin üretim aşamasında bilgisayar destekli tasarımdan (CAD) ve bilgisayar kontrollü kesim olanaklarından (CNC) yararlanılabileceğini düşünmüştür (www.cca.qc.ca/en/collection/6-greg-lynn-embryological-house).

Tasarım aşamasında Embriyolojik Ev'in geometrisinde başlangıç eğrileri ilk önce Microstation yazılımı kullanılarak elde edilmiştir. Deneyler sonucu bu belirli forma bir seri, limitleri öngörülen 12 adet kontrol noktası atanmıştır. Daha sonra Maya programına gönderilen geometrik dosya, üç boyutlu eğilme, bükülme gibi özellikler kazanmıştır ve böylece otomotiv endüstrisinde üretilenler gibi düzgün yüzeyler elde edilmeye başlanmıştır. Bu dijitalerin fiziksel modelleri tasarımın üretilebilirliğini test etmek amacıyla prototip teknolojisi kullanılarak üretilmiştir (Şekil 3.10). Modeller ışığa duyarlı üç boyutlu modeli fiziksel ortama aktararak elde edilen sakız esaslı bir reçineden veya ABS (Akrilonitril Bütadien Stiren) plastikten üretilmiştir (www.cca.qc.ca/en/collection/6-greg-lynn-embryological-house).

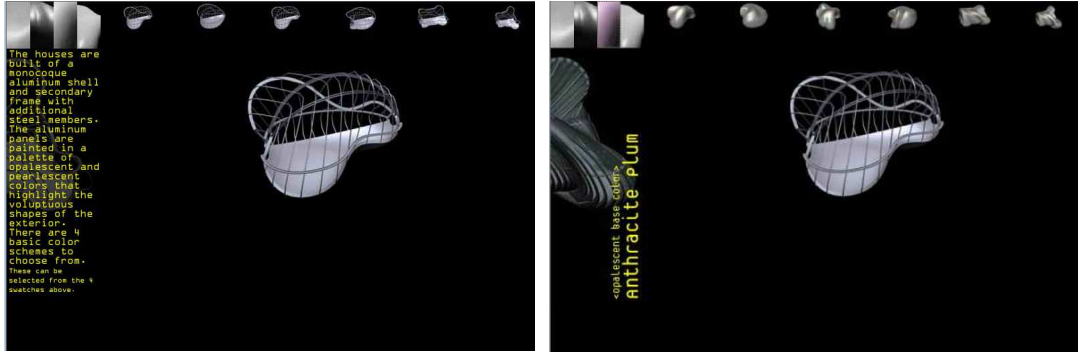


Şekil 3.10 : Embriyolojik ev projesi prototipleri (<http://www.glform.com/>).

Bu biyolojideki yumurta formundan etkilenen konut projesinin malzeme özellikleri ve bunların seçimi tasarım sürecinde önemli rol oynamaktadır. Lynn, bilinçli olarak farklı malzemeleri bilgisayar ortamındaki modelin yüzeyine atayarak, seçimi kullanıcıya bırakmaktadır. Lynn tasarladığı 6 adet birbirinden farklı konutun tasarım prensiplerini ve malzemelerin tercihini etkileşimli olarak uzman olmayan tasarımcıya web üzerinden ulaştırarak sunmaktadır (<http://www.embryological.com>). Burada örnek olarak kullanıcı ara yüzü aracılığıyla, adım adım evin gelecekteki kullanıcısına sunulan öneriler deneyimlenmiştir ve örnek ekran görüntüleri aşağıda sunulmuştur.

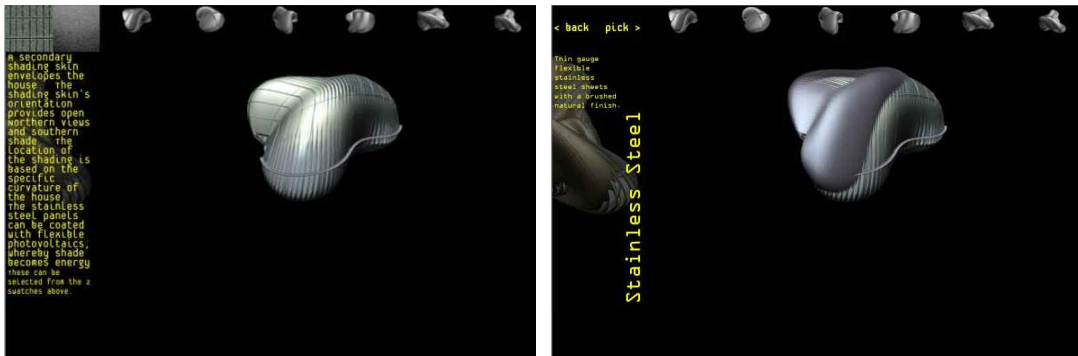
Dış strüktür olarak çelik ana taşıyıcının ve aralardaki alüminyum eğrileri kapatacak bir alüminyum monokok dış kabuk rengi seçimi yapılması için kullanıcıya dört farklı

renk sunulmaktadır. Bu renkler dış kabuğun hacimsel olarak farklı algılanması için yanardönerli renkler veya mat renklerden oluşmaktadır. Buradaki örnekte antrasit moru (anthracite plum) seçilmiştir (Şekil 3.11).



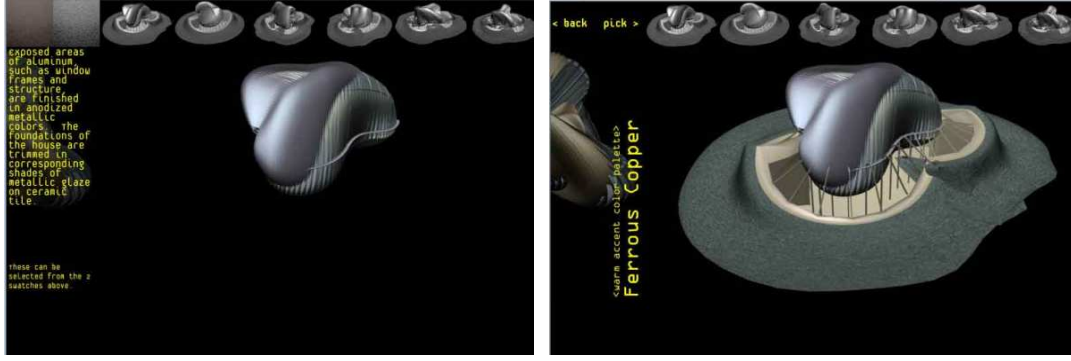
Şekil 3.11 : Embriyolojik ev projesi alüminyum dış kabuk rengi seçimi (<http://www.glform.com/>).

İkincil gölgeleme kabuğu evi sarmalar. Bu gölge veren kabuğun yerleşme düzeni, kuzey cephesini açıkta bırakacak ve güneye gölge verecek şekilde ve aynı zamanda evin eğriselliğine denk gelecek şekilde yapılandırılmıştır. Bu ikinci kabuk paslanmaz çelik olabilir veya isteğe göre paslanmaz çeliğin üzeri ince fotovoltaikler ile kaplanarak enerji üretimi sağlanabilir. Buradaki örnekte sadece paslanmaz çelik gölgeleme elemanı seçilmiştir (Şekil 3.12).



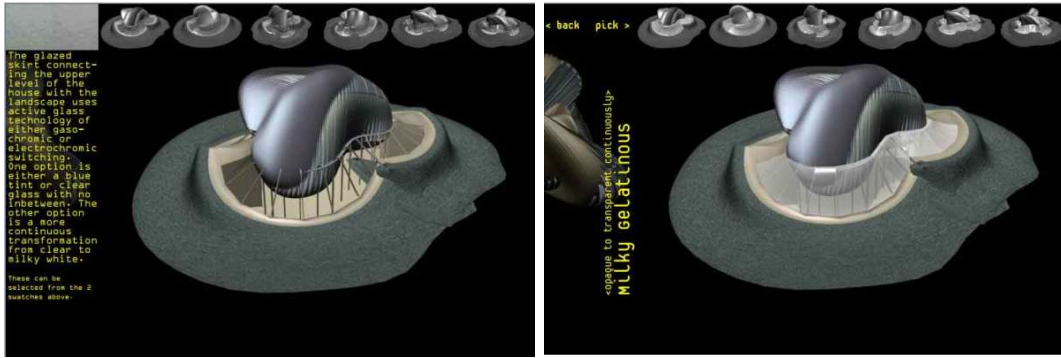
Şekil 3.12 : Embriyolojik ev projesi gölgelendirme malzemesi seçimi (<http://www.glform.com/>).

Kabuk ile topografya arasında açıkta kalacak, hava ile temas edecek pencere çerçevesi gibi bölgelerin anotlanmış metal renklerine boyanması için kullanıcıya iki renk arasından tercihi sorulur. Bu örnekte demir katkılı bakır ve cilalı alüminyumdan birincisi tercih edilmiştir (Şekil 3.13).



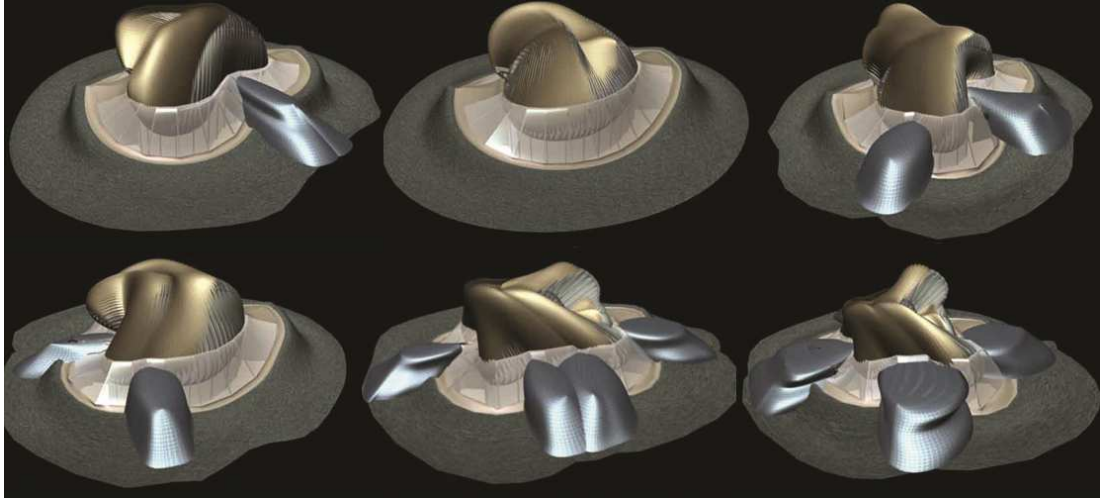
Şekil 3.13 : Embriyolojik ev projesi korozyona maruz kalacak metallerin boya tercihlerinin girilmesi (<http://www.glform.com/>).

Ev ile arazi arasında bir bağlantı kurarak ayrı mekanlar yaratan cam kaplamalarda aktif cam teknolojisi ile gazokromik veya elektrokromik malzeme kullanılması öngörülmüştür. Bu camlarda birincisi açık mavi renkli cam, diğeri ise daha fazla akıcılık sağlayan süt beyazı olmak üzere iki renk sunulmuştur. Burada süt beyazı tercih edilmiştir (Şekil 3.14).



Şekil 3.14 : Embriyolojik ev projesinde interaktif ara yüzden cam kaplama renginin tercihi (<http://www.glform.com/>).

Sonuncu tercih seçeneğinde ise bir malzeme değil, ev kütlesi ile topografya arasında bağ kuracak çıkış mekanlarının yapıya eklenmesi ile sonuçlanan 6 adet form arasından birinin seçilmesi, ara yüz aracılığı ile kullanıcıya iletilir (Şekil 3.15). Kullanıcı bu aşamada üç boyutlu olarak algıladığı bu konut formlarından kendi tercihlerine en yakın olanını seçer.



Şekil 3.15 : Embriyolojik ev projesinde sonuç formunun tercihi (<http://www.glform.com/>).



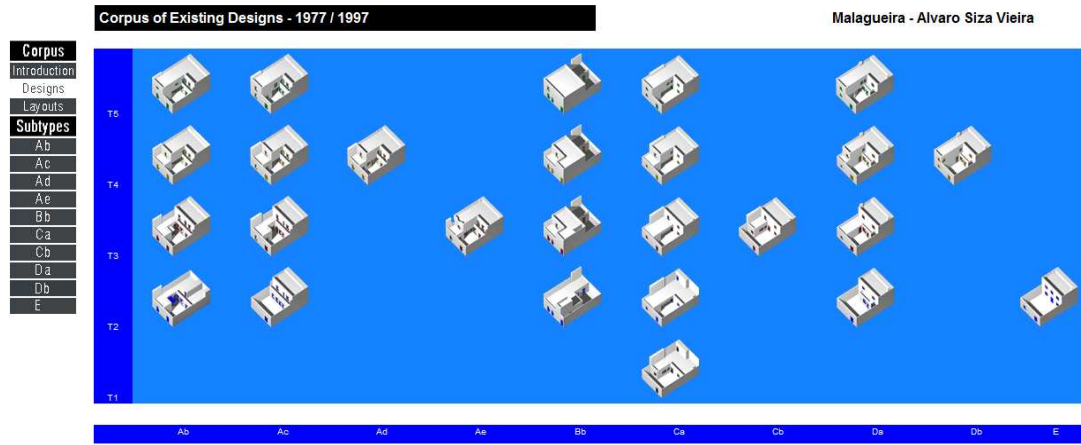
Şekil 3.16 : Embriyolojik ev projesinde XPS malzemenin CNC yardımı ile şekillenen prototip modelinin üretimi ve sergilenmesi (www.embryological.com).

Embriyolojik ev projesi inşa sürecine henüz gelmemiş olup uygulanabilirliğinin test edilmesi amacıyla, bilgisayar ortamından CNC kesim makinesine aktarılan eğrilerin bilgileri XPS malzemeler üzerine işlenerek kabuk formu birebir ölçekte prototip olarak elde edilmiştir. Daha sonra elde edilen bu eğriler bir bütün kabuk rengini temsil etmesi amacıyla boyanarak sergi mekanında sunulmuştur (Şekil 3.16).

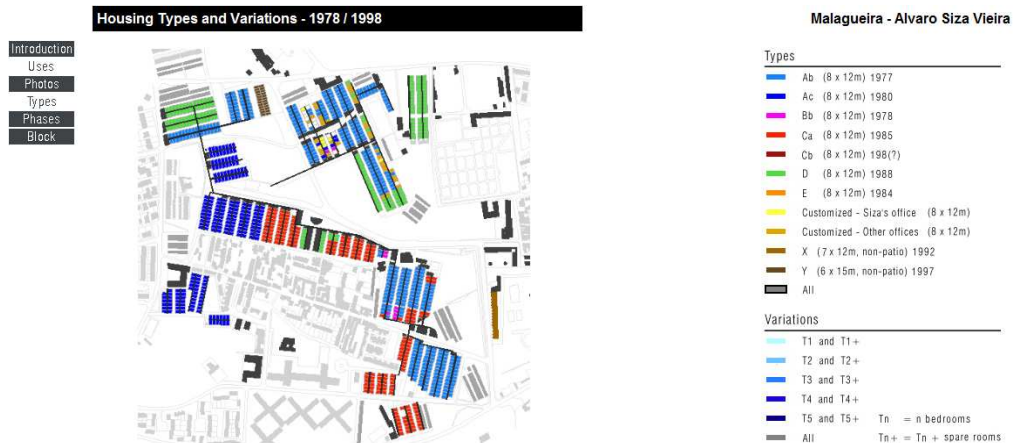
3.4 Malagueria Konutları

Bu çalışma, Portekizli mimar Alvaro Siza Vieira tarafından tasarlanan ve uygulanan Malagueria toplu konutlarının tasarım dili kullanılarak, 1999'da Mimar Jose Pinto Duarte tarafından geliştirilen, hem ikinci hem de üçüncü boyutta plan şemaları

üreten, parametrik biçim grameri ve kitlesel bireyselleştirme çalışmasıdır. 1970'li yıllarda Siza, Malagueira toplu konutları tasarımında kullanıcı katılımının daha fazla olmasını amaçlayan bir sistem geliştirmiştir. 1977 ve 1997 yılları arasında bu amaçla tasarlanan farklı özellikleri olan Siza'nın tasarım anlayışını yansıtan konutlar üretilmiştir (Şekil 3.17; Şekil 3.18) ve bu konutların tasarım sürecinde kullanıcılar tarafından özelleştirilebilmesi için birden beşe kadar farklı sayıda yatak odasına sahip, 35 farklı plan için belirli tasarım kuralları tasarlanmıştır. Fakat Siza'nın tasarladığı sistemde, konutların kitlesel bireyselleştirme potansiyeli geleneksel tasarım ve üretim araçlarının sınırlamaları nedeniyle tam olarak uygulanamamıştır (Duarte, 2005).



Şekil 3.17 : Alvaro Siza tarafından 1977-1997 yılları arasında tasarlanan mevcut Malagueira konut tasarım tipleri (<http://home.fa.utl.pt/~jduarte/malag/Malag/>).



Şekil 3.18 : Alvaro Siza tarafından 1977-1997 yılları arasında inşa edilen Malagueira konut yerleşimi ve tasarım tiplerinin gösterimi (<http://home.fa.utl.pt/~jduarte/malag/Malag/>).



Derivation of Subtype Bb, Variant T5 of the row houses for Malagueira, Portugal

Architect: Alvaro Siza Vieira
Grammar: Jose P. Duarte

Each panel shows one step of the derivation. The title is the rule used to create the block diagram and the model.

The stages and phases associated with the steps in the derivation are shown below:

Stage:	1-33	Define first floor	Phase:	9-24	Divide zones into rooms
	34-59	Define second floor		25-27	Add details
	56-61	Define terrace		28-33	Create openings
Phase:	1-2	Start floor		34	Start floor
	3-6	Locate functional zones		...	
	7-8	Define circulation scheme		61	Start floor

Şekil 3.19 : Diyagramlar Malagueira, konutlarında Alvaro Siza Vieira'nın tasarım sistemine bir konut için uygulanan tek bir biçim grameri kuralının ürettiği farklı sonuçları göstermektedir (Duarte, 2005).

Duarte, bu problemleri çözmek amacıyla Siza'nın tasarım kurallarını, biçim grameri tabanlı kurallar ile bir bilgisayar programına kodlamıştır ve Siza'nın tasarım mantığı ile tamamıyla örtüşen potansiyel bir sistem elde etmiştir (Şekil 3.19). Böylece profesyonel tasarımcı olmayan ve Alvaro Siza'nın tasarım anlayışını bilmeyen bir kullanıcı bile Siza'nın tasarım sistemine uygun konut tasarımları elde edebilecektir.

Duarte tarafından hazırlanan form bulma çalışmalarında kullanılan biçim grameri 5 farklı elemandan oluşmaktadır (Şekil 3.20). Biçim kuralları seti olan S (set of shape

rules), bilgisayar tarafından şekillerin kontrolünü sağlayan L (label), dönme, geçiş, büyütme, küçültme, yansıtma gibi basit dönüşümleri tanımlayan T (transformations), değerleri parametrelere dönüştürme fonksiyonu olan G ve son olarak başlangıç biçimi, I (initial shape) ile sembolize edilir.

Grammar

(S, L, T, G, I)

S = {Rule 1, Rule 2}

L = { • }

T = {Rotation, Translation, Scaling,
Reflection}

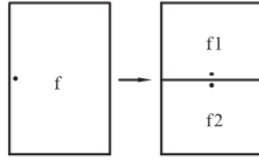
G = {g1, g2}

I = {Initial Shape}

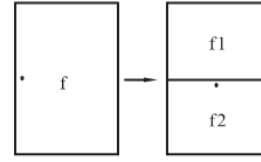
Şekil 3.20 : Malagueria konutlarında biçim grameri elemanları (<http://home.fa.utl.pt/~jduarte/malag/Malag/>).

Malagueria konutlarında uygulanan kurallardan bazıları şöyledir; A, B, C'deki kurallar, bir birimin bölünerek farklı iki birim oluşturulmasını; D, birimdeki etiketin silinmesini; E, iki birimin birleşerek bir birime dönüşmesini; F, birimlerin boyutlarının değişmesini; G, fonksiyon atanmasını; H, ise birimlerin diziliş sıralarının değişmesini tanımlamaktadır (Şekil 3.21).

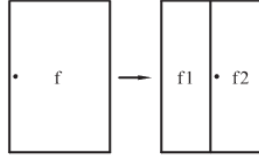
Belirlenen gramer kuralları bir kat planı için uygulanırken her oda birimi, karışıklığı önlemek amacıyla bir kısaltma ile tanımlanmaktadır. Buna göre başlangıç şekli I, iç mekan i, dış mekan o, yaşam alanı li, uyuma alanı sl, servis mekanları se, avlu ya, yatak odası be, banyo ba, mutfak ki, geçiş mekanları ts, çamaşırlık la, kiler pa, sirkulasyon alanları ci, merdivenler st ile betimlenmektedir. Belirlenen kurallar çerçevesinde bir konut planı oluşturma aşamaları Şekil 3.22'de gösterilmektedir. Burada başlangıç şekline sıra ile bazı kurallar uygulanarak apayrı kat planları oluşturulmuştur.



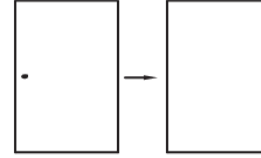
Rule A



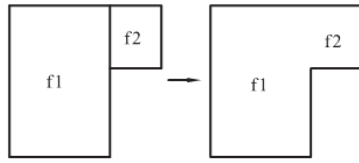
Rule B



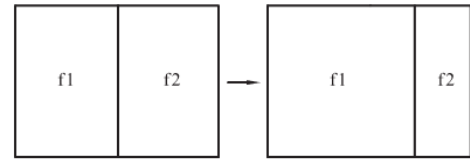
Rule C



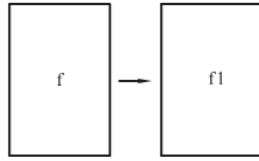
Rule D



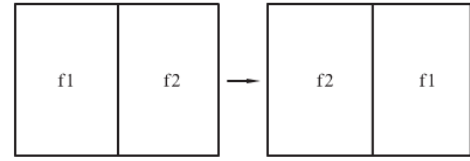
Rule E



Rule F

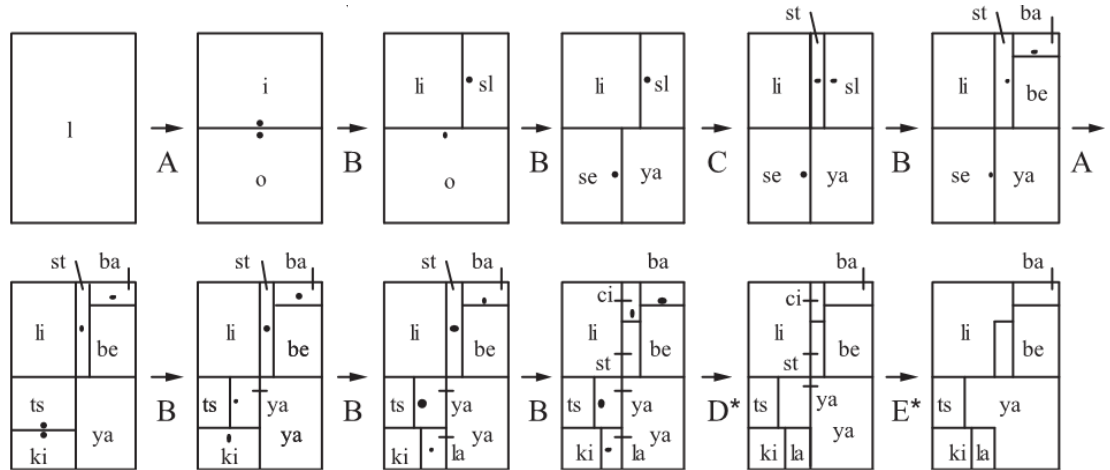


Rule G



Rule H

Şekil 3.21 : Malagueria konutlarında uygulanan biçim grameri kurallarından bazıları (Duarte, 2005).

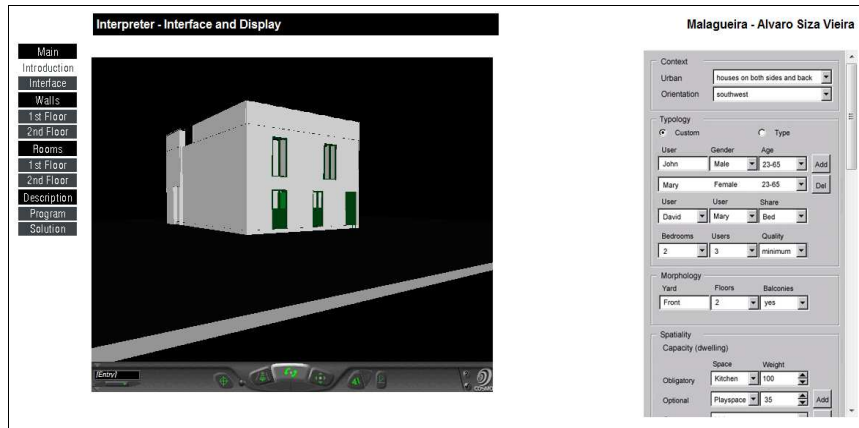


Şekil 3.22 : Malagueria konutlarında uygulanan biçim grameri ile kat planı şemaları üretiminden bir örnek (Duarte, 2005).

Duarte, bu çalışmasında Alvaro Siza'nın da en başından beri istediği gibi, kitlesel bireyselleştirme amacıyla, kullanıcıların yaşama biçimlerinin ekrana girildiği ve bunun sonucunda Siza'nın tasarım prensiplerini içeren interaktif bir arayüz

tasarlamıştır. Algoritmanın çalışması sonunda istenilen ve kullanıcılara en uygun, sonuç ürün olarak üretilen üç boyutlu ev görüntüsü kontrol edilebilmekte ve eğer değiştirilmesi istenen yerler olursa arayüzde bulunan bilgilerin olduğu kısım revize edilmektedir (Şekil 3.23). Duvarlar (Walls) başlıklı kısma gidilince evin her katının kat planları görülmekte ve odalar (Rooms) başlığında üç boyutlu kapalı ve her rengin başka bir odayı belirttiği kutular izlenmektedir. Tanımlama (Description) bölümünde kullanıcıların verdiği bilgiler doğrultusunda oluşturulan tasarım anlatılmaktadır. Tasarlanan ev hafızaya kaydedilebilmektedir ve eğer kullanıcı program tarafından üretilen sonuçtan memnun kalmaz ise, istediği parametrik değişiklikleri yaparak yeni bir konut tasarımı elde edebilmektedir.

Kullanıcı isterse evin program tarafından üretilen modeli, görsel gerçeklik ortamına taşınabilir veya model hızlı prototipleme makinasına gönderilerek üç boyutlu fiziksel bir maketi, kullanıcının tasarlanan yapıyı daha iyi ve detaylı algılaması için üretilebilir (Duarte, 1999).



Şekil 3.23 : Malagueira konutlarında kullanıcı etkileşimli konut tasarımı (<http://home.fa.utl.pt/~jduarte/malag/Malag/>).

Kitlesel bireyselleştirmeyi sağlayan bu etkileşimli programa kullanıcı web üzerinden ulaşabilmektedir. Program, bir mimarın, kullanıcılara ilk görüşme esnasında genellikle sorması gereken, aile üyelerinin profili, yaşama alışkanlıkları, istedikleri oda tipi, oda sayısı, evin morfolojisinin nasıl olmasını istedikleri, odaların boyutları, bu ev için ne kadar para harcayabilecekleri gibi sorular sorar (Şekil 3.24). Programda sorulan maliyet (cost) bölümü bir tasarım kısıtlaması olarak değil, sadece bir referans olarak konulmuştur ve programın ürettiği modeller üzerinde bir etkisi yoktur (Duarte, 2005).

Context
Urban
houses on both sides and back
Orientation
southwest

Typology
Custom
Type
User
Gender
Age
John
Male
23-65
Add
Mary
Female
23-65
Del
User
User
Share
David
Mary
Bed
Bedrooms
Users
Quality
2
3
minimum

Morphology
Yard
Floors
Balconies
Front
2
yes

Spatiality
Capacity (dwelling)
Space
Weight
Obligatory
Kitchen
100
Optional
Playspace
35
Add
Current
Living
100
Del
Capacity (spaces)
Value
Weight
Capacity
6
100
Articulation
Isolated
35
Spaciousness
Minimum
35
Width
Height
Area
4.00
2.40
16.00
Spaciousness (available and used areas)
non-useful
interior
exterior
gross
38.80
112.50
- 4.00
108.50
free
inhabitable
interior
exterior
useful
23.50
28.50
31.50
60.00
used
Ai / Au
Au / Ag
actual
minimum
actual
minimum
0.82
0.77
0.77
0.77

Topology
Rooms
Kitchen
Relation
Weight
Laundry
adjacent
90
Add
Kitchen/laundry merged
65
Del

Aesthetics
Proportion
1 : 1
30

Quality
Current
2.5 (Medium)

Weights
Function
Spatiality
Capacity
45
60
40
Articulation
30
Spaciousness
30
Topology
40
Aesthetics
55

Cost
Space Type
Cost / m2
Current cost
Covered
\$ 350.00
\$37,500.00

Program
Done
Save
Load
Solution
Derive
Save
Load

Şekil 3.24 : Malaguera konutlarında kullanıcı ile etkileşime geçen bilgi motoru (<http://home.fa.utl.pt/~jduarte/malag/Malag/>).

Duarte'nin arařtırıp elde ettiđi biçim grameri ile, belki de Alvaro Siza'nın kendisinin bile zaman kısıtlaması nedeniyle tasarlayamayacađı konutları, artık bu kullanıcı etkileřimli programı kullanan herkes istediđi özelliklere ve Siza'nın tasarım anlayıřına sahip, sadece o kullanıcıya özel olacak konut üretilebilecektir. Ayrıca konut tasarımları, uzman tasarımcı olmayan kullanıcının daha iyi algılayabilmesi için sadece iki boyutlu deđil, üçüncü boyutta da sonuç üretmektedir.

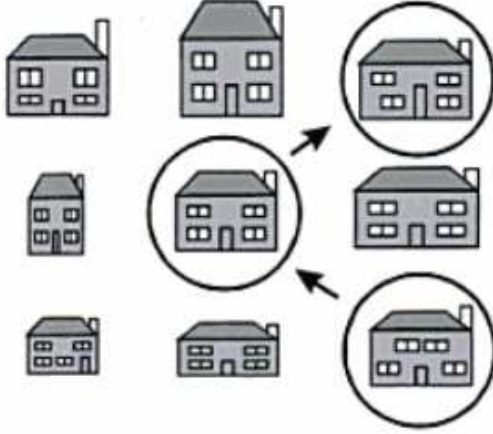
4. KİTLESEL BİREYSELLEŞTİRME AMAÇLI EVRİMSEL YAKLAŞIM

Evrimsel tasarım çalışmalarının bir parçası olan ve gelişen teknoloji ile yapay zeka çalışmalarında da sıklıkla başvurulmuş Genetik Algoritmalar, bir problemi evrimsel süreçten geçirerek, optimum çözüm üreten bir bilgi işleme tekniğidir. Bu tez çalışması kapsamında, kullanıcı (uzman tasarımcı olmayan kullanıcı) odaklı, kitlesel bireyselleştirmeyi esas alan, geleneksel yöntemlerle çözüme ulaşılması zahmetli ve karmaşık olan tasarım problemlerinde Genetik Algoritmaların kat planı üretimi ve form arama çalışmasında kullanılması bir öneri olarak sunulmuştur.

Evrimsel tasarımın özünü oluşturan Genetik Algoritmalar, doğadaki evrim, doğal seçim, ve üreme süreçlerini sanal ortamda taklit ederek çalışan bir arama ve en iyileme yöntemidir (Emel & Taşkın, 2002). Genetik Algoritmalar, diğer tasarım çalışmaları gibi bir problemin çözümünü doğrusal hesaplama yollarındaki öğrenme metoduyla bilinçli bir şekilde yürütmek yerine, rastlantısal kazanımların da dahil olduğu doğal bir arama sürecinde belirli kısıtlamaların yön vermesi sayesinde yaratıcı, şaşırtıcı tasarımlara olanak sağlar.

4.1 Evrimsel Bilgi İşleme

Evrimsel tasarımla bilgi işleme tamamen arama (search) sistemine bağlı olarak çalışmaktadır. Bilgisayar bilimindeki arama algoritmaları, bir problemdeki bütün olası çözümleri içeren bir arama uzayını (search space) ve bu uzaydaki bir noktada çözümün tanımlandığı bilgi işlemeyi tanımlar (Kanal & Cumar, 1988). Bir uygulamadaki gelişen parametrik değerler, daha sonra çözüm uzayında (solution space) daha iyi sonuçlar elde etmek amacıyla problemin arama uzayına aktarılır. Arama uzayında parametrik değerleri birbirini takip eden çözümler elde edilir (Şekil 4.1). Bilimsel çalışmalarda kullanılan birçok arama algoritması vardır ve evrimsel arama algoritması bunların arasında günümüzde en geçerli olanı ve hızla gelişenidir (Bentley, 1999).



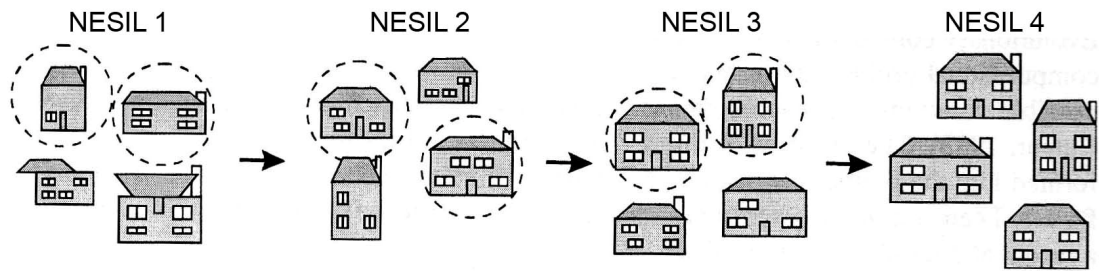
Şekil 4.1 : Örnek bir ev tasarımına arama uzayında parametrik değerlerle çözüm aranması (Bentley, 1999).

Teknoloji ile farklılaşan yeni parametrelere ve karmaşık ortam koşullarına göre bir proje farklı aşamalar içerebilir. Farklı veri girişleri, tasarım yöntemi ve çözümleri her projeye göre değişkenlik gösterebilir. Bu tasarım aşamaları bazen projedeki en iyileme problemleri olarak da düşünülebilir. Analitik veya sayısal metotlar bir fonksiyondaki en yüksek ve en alttaki değerleri hesaplamak için uzun zamandan beri kullanılmaktadır ve bunlar bazı problemlerde iyi sonuç verirken daha karmaşık problemlerde sonuçsuz kalabilmektedir. Evrimsel Bilgi İşleme doğadaki evrimden esinlenilerek ortaya çıkmıştır. Bu tür algoritmalar doğal evrimsel analojiyi kullanarak problemlerin çözümlerini evrimleştirerek bir arama yaparlar (Bentley, 1999). Evrimsel algoritmalar özellikle karmaşık optimizasyon sorunları üzerinde etkin bir arama metodu içerir. Mimari tasarımdaki gibi gerçek tasarım problemlerinde, parametreler birden fazla olabilir ve etki ettikleri optimizasyon değeri çok karmaşık ve doğrusal olmayan bir yapıda olabilir. Böyle karmaşık problemler klasik çözüm yollarıyla aşılamaz veya sadece sınırlı bir değerde kalırlar. Fakat bu tür problemler, Genetik Algoritmalar gibi problemlere global olarak en uygun çözümü bulabilen evrimsel algoritmalar içeren olasılıklı optimizasyon teknikleri ile kabul edilebilir nitelikte çözüme kavuşabilir. Evrimsel algoritmalar, mantık olarak diğer analitik arama metotlarına göre farklıdır. Ana farklılık, diğer bilgisayar destekli çözüm yolları arama alanındaki bir noktayı işlerken, evrimsel algoritmalar potansiyel çözümlerin popülasyonunu işler. Böylece tasarımda bir noktada çözüm aramak

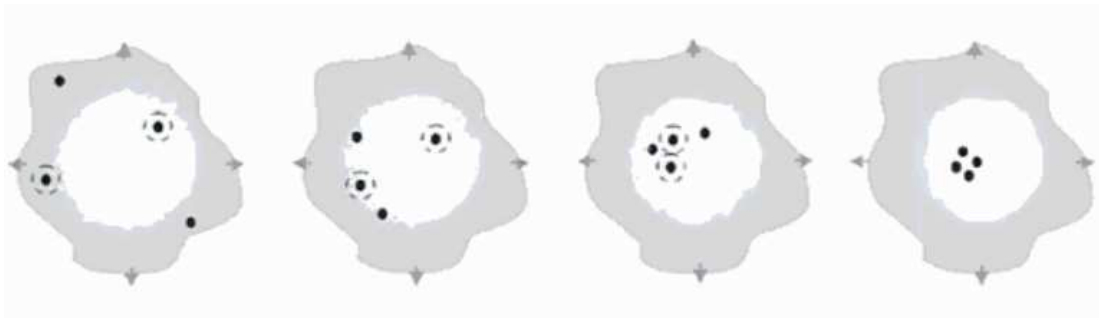
yerine, tüm olasılıklı çözüm önerilerinin içerisinde verilen parametrelere ve sınırlandırmalara göre en uygun sonuç nesil elde edilir.

Evrimsel algoritmalarda, bilgisayar çözümlerin popülasyonundan bir dizi elde eder, daha sonra bu popülasyonda daha iyi sonuçların ‘çocuklarının olmasına’ ve daha kötü sonuçların ‘ölmesine’ izin verir. Çocuk çözümler karakter olarak bazı rastgele farklılıklarla atalarına benzer ve daha sonra bu çocuklardan aralarında yeterince iyi olmayanlarının ölmesine karar verirken, iyi olanlarının çocuklarının olmasına izin verir ve bu algoritma sonlandırılana kadar böyle devam eder. Bu basit işleyiş evrimleşmenin oluşmasına yol açar, bilgisayar belirli sayıdaki nesil üretiminden sonra başlangıçtaki nesillere göre büyük ölçüde daha kaliteli sonuçlar elde eder (Bentley, 1999).

Arama uzayı göz önünde bulundurulursa, evrimin nasıl iyi sonuçlar bulacağı hakkında bilgi sahibi olunabilir. Şekil 4.2 popülasyon sayısı dört olarak belirlenen konut tasarımlarını, Şekil 4.3 bu konutların arama uzayını örneklemektedir.



Şekil 4.2 : Popülasyon sayısı dört olarak evrimleşen konut tasarımlarının dört nesli (Gelecek neslin ebeveynleri yuvarlak içine alınanlardır) (Bentley, 1999).



Şekil 4.3 : Evrimleşen konut tasarımlarının her nesilde arama uzayındaki yerleri (Bentley, 1999).

Günümüzde dört farklı evrimsel bilgi işleme türü kullanılmaktadır. Bunlar; Evrim Stratejileri, Evrimsel Programlama, Genetik Programlama ve Genetik Algoritmalar. Bu tez çalışmasının model tasarımında ağırlıklı olarak Genetik Algoritmalar kullanıldığı için Genetik Algoritmalarla daha detaylı bir anlatımla yer verilecektir.

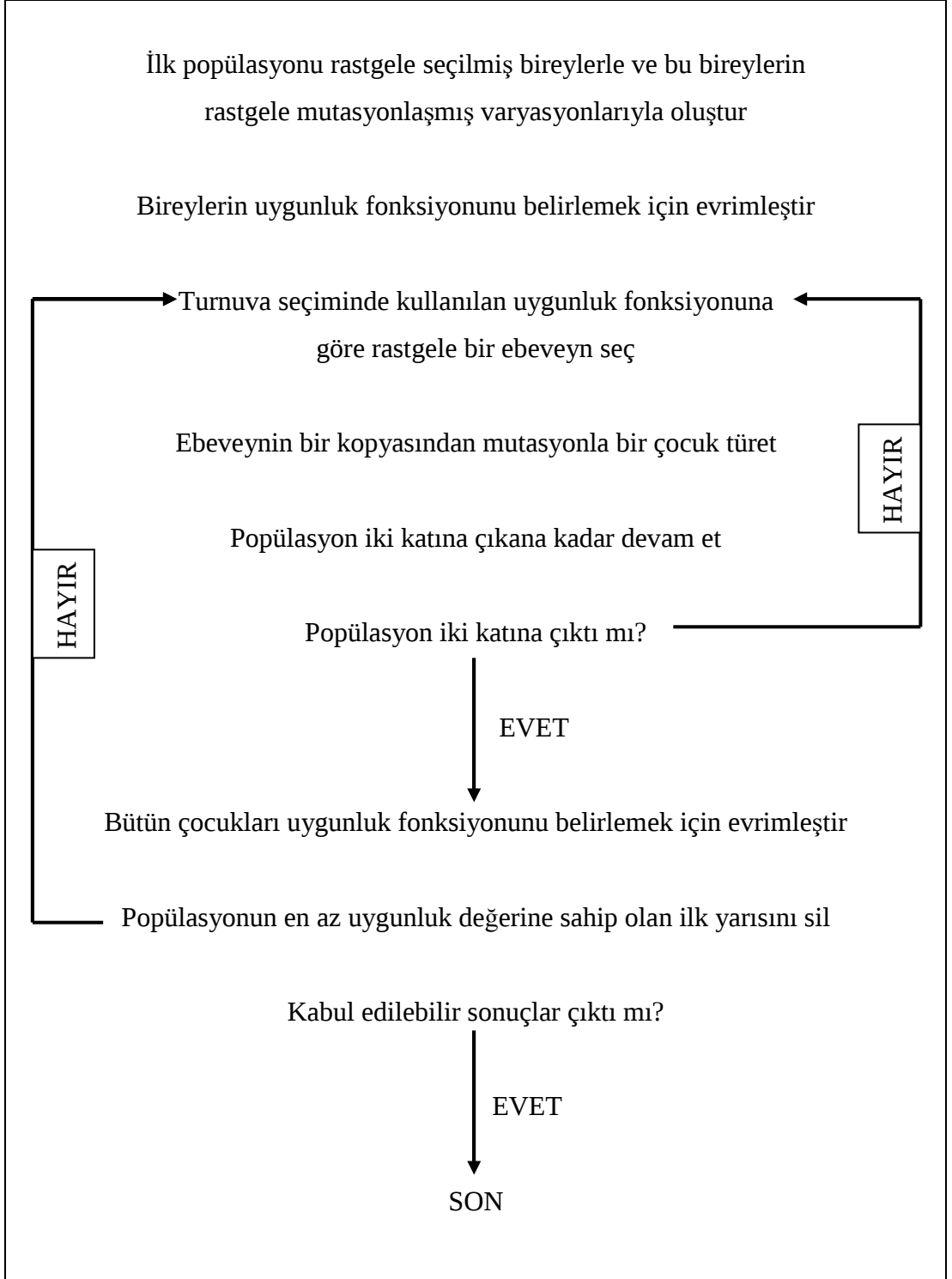
4.1.1 Evrim stratejileri

Evrimsel Stratejileri (ES) ilk olarak 1960’larda I. Rechenberg tarafından “Evolutionstrategie” isimli eserinde tanıtılmıştır (Rechenberg, 1973). Bu stratejik yaklaşım üremenin eşleştirmeye dayalı olarak değil, denetimli bir değişikliğe uğratmaya dayalı olarak yapıldığı, doğrusal veri yapılı, genlerin yalnızca sayısal değerlerden oluştuğu bir yöntemdir (Buradaki sayısal kavramı genin değerini değiştirmek bakımından aritmetik işlemlere tabi tutulabilmesi anlamındadır). Üremede her bireyin her bir gen değeri yine, evrimsel yöntemle değiştirilen ve o bireyin o genine özgü olan bir standart sapmanın belirlediği bir normal olasılık dağılımına göre farklılığa uğratılır (Üçoluk, 2002).

4.1.2 Evrimsel programlama

Evrimsel Programlama (EP), kuramsal mantığı ile evrimsel stratejilere benzer, fakat ES’den daha önce ve bağımsız olarak Lawrence Fogel tarafından 1960’larda geliştirilmiştir. Evrimsel programlamada tek bir popülasyonun sonuçları hedeflenir ve yeniden üretim aşaması sadece mutasyon kullanılarak yapılır (Fogel, 1966). EP’da üretim ve seçim yöntemi mutasyon ve rekabettir, Genetik Algoritmalarla ise mutasyon ve genetik değişimdir. EP’de işlem zamanından önemli ölçüde tasarruf sağlanmaktadır (Sinha, Chakrabarti and Chattopadhyay, 2003).

Evrimsel Programlamanın algoritması Şekil 4.4’de görülmektedir.



Şekil 4.4 : Evrimsel programlamanın akış diyagramı (Bentley, 1999).

Evrimsel Programlama da, ES de olduğu gibi doğrudan problemin veri değişkenleri üzerinden işlem yapar ve arama uzayı ile çözüm uzayı aynıdır.

4.1.3 Genetik Programlama

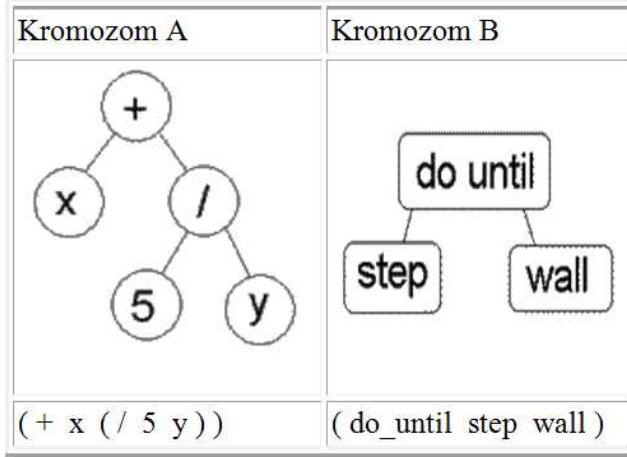
Genetik Programlama (GP), Genetik Algoritmaların bir uzantısı olarak kabul edilen, özelleştirilerek uygulanan, ele alınan problemin yapı taşlarından oluşturulan muhtemel ilkel çözüm tarzlarının belli bir uyum kriterine göre evrimleşerek en iyileşmesini amaçlayan bir evrimsel algoritma tekniğidir (Koza, 1992). Genetik Programlamada amaç evrim teorisinden esinlenerek oluşturulan Genetik Algoritmalar kullanılarak, kullanıcı tarafından tanımlanmış görevlerin yerine getirilmesidir (Hatunoğlu, 2010).

1985 yılında ise Michael L. Cramer tarafından ilk kez “ağaç tabanlı genetik programlama” ortaya atılmış, ve bu daha sonra John R. Koza tarafından geliştirilerek birçok karmaşık en iyileme ve arama problemlerinde kullanılmıştır. 1990’lı yıllarda ise bilgisayarlardaki artan işlemci gücüyle beraber bu yeni bilim kolu olan genetik programlamanın kullanım alanı daha da genişlemiş; bu yeni metod kuantum hesaplamaları, elektronik tasarım, oyun programlama, sıralama, arama gibi birçok alanda yoğun bir şekilde kullanılmaya devam edilmiştir (Hatunoğlu, 2010). Daha sonra GP programcılar tarafından da yüksek kavramsal kapasitesinden dolayı benimsenerek literatürde Genetik Algoritmaların yanında yer almaya başlamıştır. Genetik Programlama, GA da olduğu gibi temelde aynı evrimsel çalışma prensibine sahiptir ve bireylerden oluşan popülasyonlar amaçlanır. Fakat burada arama ve sonuç uzayı gibi ayrı bölge tanımlanmaz, genotipler ile fenotipler aynıdır.

GP’ de bireyleri oluşturabilmek için iki farklı sınıftan gelen nesneler kullanılır. Bu nesneler terminaller ve fonksiyonlardır. Terminaller problemin tanımlanmasında doğrudan etkin olan değişkenler veya sabitlerdir. Fonksiyonlar ise terminalleri belli bir işlem mantığına göre birleştirmeye yarayan nesnelerdir. Fonksiyonlar ve terminallerin hiyerarşik bir anlayış içinde anlamlı bir yapı oluşturacak şekilde birleştirilmesi sonucunda bireyler oluşur (Langdon & Poli, 2002). Oluşan bireylerin boyutu derinlik olarak ifade edilen ve içerisindeki fonksiyon sayılarının toplanması ile oluşan bir büyüklükle ölçülür (Göloğlu & Arslan, 2006).

Genetik programlama ve ağaç kodlama metodu: Ağaç kodlama temel olarak programları veya etkilerini evrimleştirmek için kullanılmaktadır. Ağaç kodlamada, fonksiyonlar veya programlama dilindeki komutlar gibi her kromozom bazı nesnelerin ağacıdır. (Şekil 4.5.). Ağaç kodlama evrimleşen programlar veya ağaç

sekinde kodlanabilecek herhangi diğerk yapılar için uygundur. LISP yazılım dilinde, programların ağaç sekinde temsil edilmesi sebebiyle LISP genetik programlama için en çok kullanılan dildir. LISP'te bu ağaçlar kolayca ayrıştırılıp, çaprazlama ve mutasyon kolayca yapılmaktadır.



Şekil 4.5 : Ağaç kodlama ile kodlanmış kromozomların örneği (Obitko,1998).

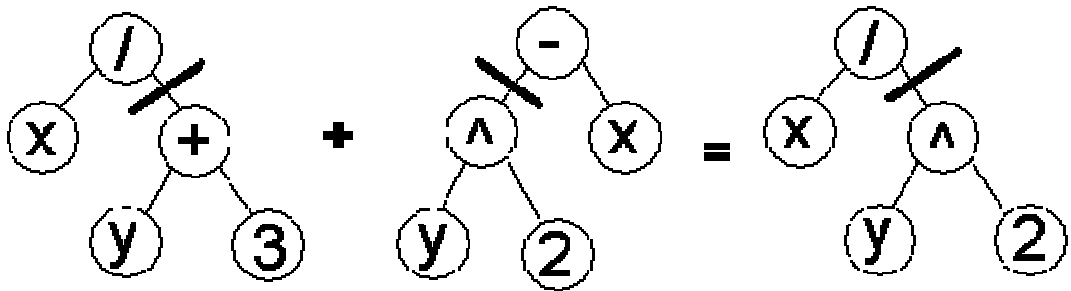
Ağaç kodlama Örneği: Verilen değerlerden fonksiyonu bulma

Problem: Bazı giriş ve çıkış değerleri verilmiştir. Görev bütün girdilere en iyi çıkış değerini (istenilene en yakın değer) veren fonksiyonu bulmaktır.

Kodlama: Kromozomlar ağaçtaki fonksiyonlar olarak temsil edilmektedir.

Ağaç Çaprazlaması: Her iki ebeveynde de bir çaprazlama noktası seçilir, ebeveynler bu noktadan bölünürler ve çaprazlama noktasının altında kalan parçaları yer değiştirilerek yeni çocuk kromozomları oluştururlar (Şekil 4.6).

Mutasyon: (operatör veya numaraların mutasyonu) Seçilen düğümler yer değiştirir (Obitko, 1998).



Şekil 4.6 : Ağaç kodlamada ebeveyn kromozomların bir noktadan kesilerek çaprazlanması (Obitko, 1998).

4.1.4 Genetik Algoritmalar

Genetik Algoritmalar; doğal seleksiyon prensibinden yola çıkılarak geliştirilmiş doğadaki evrime dayanan güçlü ve etkili araştırma algoritmalarıdır (Holland, 1975). Bilgisayara doğayı taklit etme özelliğinin öğretildiği ve evrimsel hesaplama tekniğinin bir parçası olan Genetik Algoritmalar (GA) John Holland tarafından 1975'te yayınlanan "Adaptation in Natural and Artificial Systems" kitabı ile tanıtılmıştır (Holland, 1975). İlk olarak Holland doğal evrim sürecini Genetik Algoritmalar içinde en iyileme problemleri için kullanmıştır. Daha sonra David Goldberg isimli öğrencisi, tezinde gaz boru hattının kontrolünü içeren bir problemin çözümünü GA ile bulmuştur (Michalewicz, 1996).

Son yıllardaki geniş bir zaman aralığı süresince en iyileme (optimizasyon) çalışmalarında çokça kullanılan GA, artık "insan araştırmalarındaki deneyimsel içgüdülerin de kısmen yer aldığı arama algoritmaları" olarak da tanımlanmaktadır (Goldberg, 1989). GA aynı zamanda kötü uygulanmış, kısıtlı olarak işlenmiş bile olsalar, yine de kabul edilebilir sonuçlar üreten hata kaldıran algoritmalarıdır (Davis, 1991). Günümüzde GA, geniş çeşitliliğe sahip optimizasyon problemlerinin üstesinden gelebilmektedir ve bu gelişmeyle doğru orantılı olarak mükemmele yakın sonuçlar sağlanabilmektedir (Goldberg, 1989; Davis, 1991).

Holland, doğadaki adaptasyon süreci ile yeni yapay sistemlerin adaptasyon sürecinin benzer işleyiş temeline dayandığını ve problemlere çözüm arayışının doğal evrimsel süreçle paralel çalıştığını belirtmiştir. Buna göre doğada yeni organizmalar çevrelerindeki gelişime evrimleşerek uyum sağlayabilmektedirler. GA da verilen problemin çözümlerini geliştirerek benzer şekilde sonuca ulaştırmaktadır. Amaç bu çözümlerin toplamından bağımsız bireyler popülasyonunu oluşturmak ve çoklu arama havuzuna eklemektir. Burada birey olarak belirtilenler genlerin oluşturduğu kromozomlardır. Evrimsel süreçte bireyler kalıtsal özelliklerini gösteren kromozomlar ile kontrol edilirler. Doğanın analogisi gereği, problemin en iyi sonuca ulaşması için potansiyel çözümlere uygun olarak, bireylerden yaşam ortamına uygun olanları yeniden üretilmek üzere saklanır, zayıf ve yaşamak için yeterince elverişli olmayanları ölür. Yeni bireyler bu saklanan bir veya iki ebeveynin çaprazlaması ve mutasyonu sonucunda yaratılır. Ebeveynleri ile benzer özellik gösteren bu yeni bireyler, nesildeki eski bireylerin yerini alır. Sonuç olarak, yeni bir jenerasyonun

bireyleri ile eski nesilden uygunluk şartlarını yerine getiren bireyler arasında benzerlikler vardır.

Genetik Algoritmalar aynı zamanda bahsedilen evrimsel süreçleri kullanan bir makine öğrenmesi (machine-learning) modelidir. Bu modelde, makine içerisinde belli kromozomlarla temsil edilen bireylerden oluşan bir topluluk yaratılır. Bu bireyler aslında canlılardaki base-4 biçimindeki DNA'dan oluşan kromozomlara eşdeğer karakter dizileridir. Bu bireyler simüle edilmiş bir evrim sürecinden geçerler. GA'lar birçok değişik alanda uygulanabilirler; örneğin çok boyutlu bir optimizasyon probleminde, karakter dizileri optimize edilmek istenen sistemin değişik parametrelerinin değerlerini temsil edebilirler. Pratikte, bu karakter dizileri genetik modeldeki kromozomlar gibi kabul edilebilir. Basit bit değişimleriyle kolayca mutasyon, yeniden-birleşme ya da başka işlemleri gerçekleştirebilir. Bir GA şu sırayı takip eder:

- i) Topluluktaki bireylerin güçlülüğünü değerlendir.
- ii) Yeniden birleşim, güçlülüğe dayalı üreme ve mutasyon işlemlerini kullanarak yeni bireylerden oluşan topluluk yarat.
- iii) Eski bireyleri ele ve yeni topluluğa aynı işlemi uygula.

Bu döngünün her yinelenmesi yeni bir kuşağı temsil eder. Kuşak(0) rastgele bir topluluktur, fakat ilerleyen kuşaklar (1....n) genetik süreçler sonucu gelişme gösterirler (Atalağ, 2001).

Yapay evrimsel sürecin birer temsilcisi olan Genetik Algoritmalar da bu doğal süreci taklit eder. Amaç bir nesli oluşturacak sonuç elde etmek, bu nesildeki uygun bireylerin yeni bireyler oluşturmaları için izin vermek ve daha az uygun olanları sistem dışı bırakmaktır. Yeni bireyler ebeveynlerine benzer genetik özellikler taşımaktadır ve en uygunları yeni nesle taşınmak üzere hayatta kalabilmektedir. Sonuç çözümler, geçmiş nesillerdeki atalarından çok daha iyi olacaktır.

4.1.4.1 Genetik Algoritmaların temel teoremi

Genetik Algoritmaların arama uzayında nasıl arama yaptığı alt dizi kavramıyla açıklanmaktadır. Alt diziler, GA'ın davranışlarını açıklamak için kullanılan teorik yapılardır. Bir alt dizi, belirli dizi kümeleri arasındaki benzerliği tanımlayan bir dizidir. Alt diziler, {0 , 1 , *} alfabesi kullanılarak tanımlanır. Örneğin H alt dizisi,

ilk konumunda 0, ikinci ve dördüncü konumunda 1 değeri olan kromozomlar kümesi içindir.

$$H = 0 \ 1 \ * \ 1 \ *$$

* sembolü dizinin o konumunun hangi değeri aldığına önemli olmadığı anlamındadır. Dizi o konumda 0 veya 1 değeri alabilir. Eğer bir x dizisi, alt dizinin kalıbına uyarsa x dizisine “H’nin bir örneğidir” denir. Alt dizilerin iki özelliği mevcuttur (Emel & Taşkın, 2002). Bu özellikler şunlardır:

Alt dizi derecesi: Bir H alt dizisinin derecesi $o(H)$ ile gösterilir ve mevcut alt dizi kalıbında bulunan sabit konumların sayısıdır. Bu sayı ikili alfabede 0 ve 1 değerlerinin sayısının toplamına eşittir (Goldberg, 1989).

Alt dizi uzunluğu: Bir H alt dizisinin uzunluğu $\delta(H)$ ile gösterilir ve mevcut alt dizi kalıbında bulunan belirli ilk ve son konumlar arasındaki uzaklıktır (Goldberg, 1989).

Alt dizi derecesi ve alt dizi uzunluğu kavramlarının Genetik Algoritmaların temel teoreminde son derece önemli bir yeri vardır. Alt dizi derecesi düşük, alt dizi uzunluğu kısa olan diziler “yapı blokları” olarak adlandırılır. John Holland, Genetik Algoritmaların işleyişinde uygun yapı bloklarının tanımlanmasını ve bu yapı bloklarının daha uygun yapı blokları elde etmek amacıyla birleştirilmesini önermektedir. Bu fikir yapı blokları hipotezi olarak bilinmektedir (Emel & Taşkın, 2002).

Genetik Algoritmaların temel teoremi ise Yeniay tarafından şöyle açıklanmaktadır:

Popülasyon ortalamasının üstünde uyum gücü gösteren, kısa uzunluğa ve düşük dereceye sahip alt diziler zamanın ilerlemesiyle üstsel olarak çoğalırlar. Bu çoğalma, genetik işlemler aracılığı ile gerçekleşmektedir ve sonucunda ana-babadan daha üstün özellikler taşıyan bireyler ortaya çıkmaktadır (Yeniay, 2001). Sonuç uzayındaki bireylerin önceki nesillere göre kalitelerinin daha yüksek olması iki şekilde açıklanmaktadır. Öncelikle başarısız olan bireylerin üreme şansları azaltıldığı için kötüye gidiş zorlaşmaktadır. İkinci olarak GA’nın yapısı kötüye gidişi engellemekle kalmamakta, Genetik Algoritmaların temel teoremi uyarınca, zaman içinde hızlı bir iyiye gidiş de sağlayabilmektedir (İşlier, 2001). Genetik Algoritmalar doğadan esinli yapıları gereği, bir döngü oluşturarak uygun olmayan bireyleri seçim işlemcileri ile elemektedir. Bu döngü sonlandırma kriteri sağlanana kadar en iyi çözüm aralığını aramaya devam eder.

4.1.4.2 Genetik Algoritma kavramları

Doğal evrimsel süreç ile Genetik Algoritmaların çalışma mekanizmalarının aynı olduğu gibi, kullanılan terminolojileri de benzer özellikler göstermektedir. Bunlardan temel olanları şöyledir:

Kromozom

Genetik Algoritmalarda bir çözüm uzayındaki her nokta, kromozom (birey) adı verilen genellikle dokuz bitlik ikili kodlama dizisi ile kodlanır. Karakter dizilimindeki her ikili bit bir çözümün özelliğini temsil eder (Bentley, 1999). Böyle ikili kodlanan kromozomlar şöyle görünmektedir:

Kromozom 1: 1101100100110110

Kromozom 2: 1101111000011110

Gen

Kromozom içerisinde bulunan ve organizmanın karakterlerinin tayininde rol oynayan kalıtsal birimlere gen adı verilir. Doğadan esinlenen yapay sistemlerde gen, kendi başına anlamlı bilgi taşıyan en küçük birim olarak tanımlanmaktadır.

Popülasyon

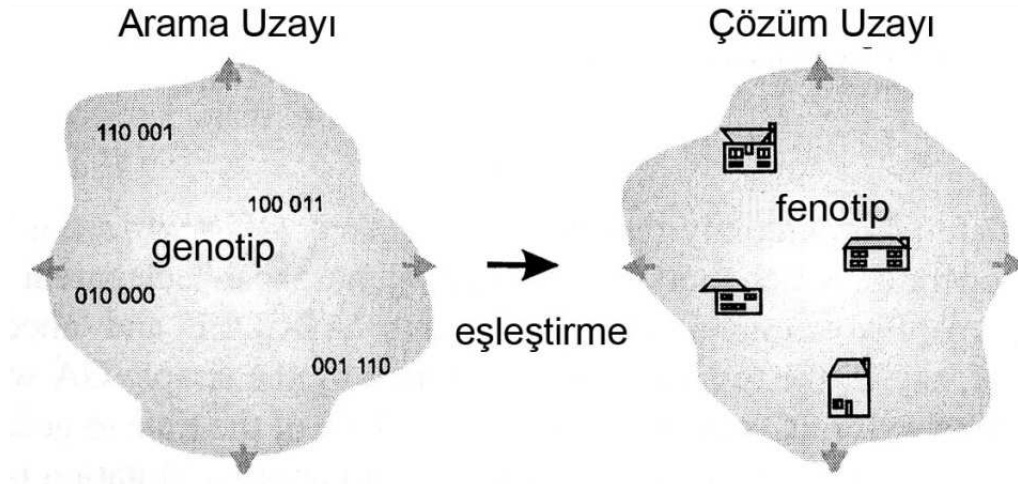
Popülasyon, geçerli alternatif çözüm adayları kümesini içeren kromozomlardan oluşan topluluğa denir. Popülasyondaki her birey geçerli bir çözümü içerdiği halde farklı çözüm seviyesine sahiptir ve bir popülasyondaki birey sayısı (kromozom) genellikle algoritma süresince sabit tutulur. GA’da popülasyonda varolan birey sayısı ile ilgili genel bir kısıtlama yoktur, birey sayısına deneme yanılma yoluyla karar verilebilir. Popülasyondaki kromozom sayısı arttıkça çözüme ulaşma süresi (iterasyon sayısı) azalır.

Genotip (G)

Bireylerin sahip olduğu genlerin toplamına genotip adı verilir. Bireyin genotipini oluşturan genler o bireyin kendine özgü karakterlerinin meydana gelmesinden bütünüyle sorumludurlar. Genetik kod adı verilen belirli bir dilde yazılırlar. Genotipler arama uzayındaki parametrelerin birer kodlanmış versiyonudur.

Fenotip (F)

Bir bireyi oluşturan karakterlerin bütününe fenotip adı verilir. Fenotipler, çözüm uzayındaki gerçek çözümlerin dışarıya yansımasıdır ve doğal işleyişte bir organizmadaki genlerin çevresel etkenlerin etkisi altında fiziksel olarak gözlemlenen sonuçlarıdır. Bir popülasyondaki her birey genotiplerden ve bunlara karşılık gelen fenotiplerden oluşur. Şekil 4.7’de görüldüğü gibi genotipler arama uzayında, çözüm uzayında bulunan fenotiplerin karakteristik özelliklerinin kodlanmış halini simgelemektedir. Fenotipler bir ev örneğinde parametrik olarak evin çatısı, pencere sayısı, yeri, kapısının ve bacasının yeri, evin boyutları gibi fiziksel özellikleri dışa yansıtırken, genotipler bu dışa yansıyan özelliklerin her birinin parametrik yazılımlarını tanımlar (Bentley, 1999).

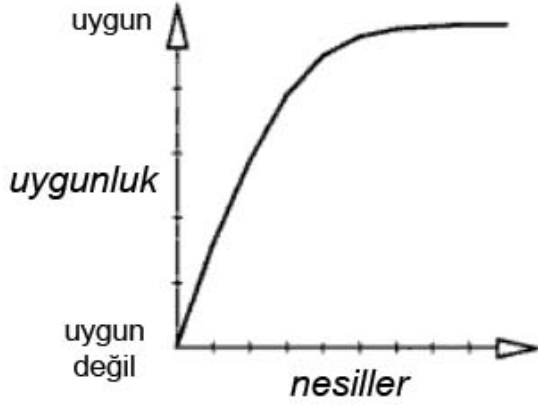


Şekil 4.7 : Arama uzayındaki genotipler ile çözüm uzayındaki fenotiplerin eşleşmesi (Bentley, 1999).

Uygunluk fonksiyonu

Arama uzayında en iyi sonuçlara varılabilmesi için popülasyondaki her bireyin bir sonraki nesle aktarılıp aktarılamayacağı belirli kısıtlamalar dahilinde olmalıdır. Tersini düşünülürse rastgele gelecek nesillere aktarılan bireylerden sağlıklı çözümler elde edilemez. Uygunluk Fonksiyonu (Fitness Function) her popülasyonun amaç değeridir. Bu fonksiyon ile, kromozom içerisindeki kodlanmış ya da kodlanmamış bilgiler çözümlenerek sayısal bir uygunluk değeri elde edilir. Bu bilgi, her kuşakta daha uygun çözümlerin seçiminde yol göstermektedir. Bir çözümün uygunluk değeri ne kadar yüksekse, yaşama ve çoğalma şansı o kadar fazladır ve bir sonraki kuşakta

temsil edilme oranı da o kadar yüksektir (Yeniay,Ö., 2001). Uygunluk değerindeki oynamalarla sonuç ürünlerin iyi olup olmaması sağlanabilir. Nesiller geliştikçe popülasyondaki bireyler, eski nesillerdeki bireylere oranla uygunluk değerine daha yakınsar (Şekil 4.8).



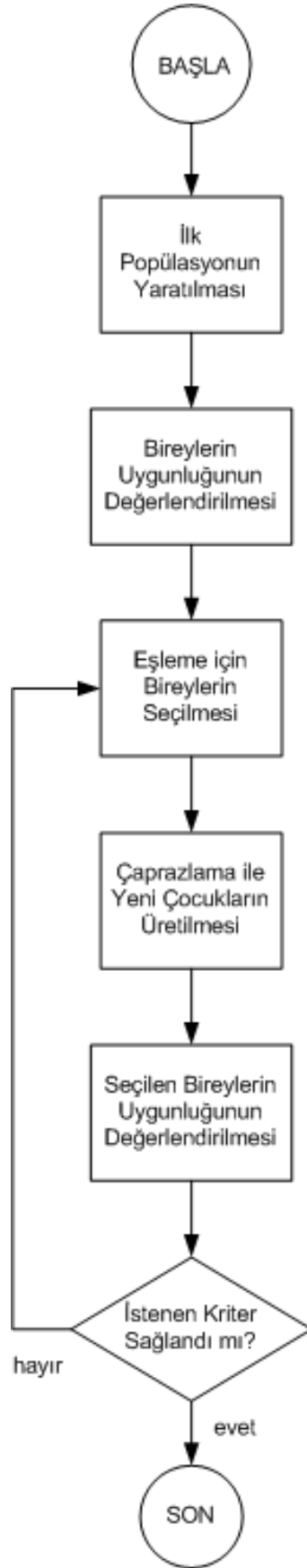
Şekil 4.8 : Nesiller geliştikçe değişen uygunluk değeri (Bentley, 1999).

4.1.4.3 Genetik Algoritmaların çalışma prensibi

Genetik Algoritmalar Darwin'in evrim teorisinden esinlenilerek ortaya çıkmış olan ve karşılaşılan problemlere evrimsel süreç kullanılarak çözüm arayan sayısal optimizasyon yöntemleridir. GA tek bir çözüm noktası üzerinden değil, çözümlerin popülasyonu üzerinden işlem yaparlar. Bir popülasyondan alınan sonuçlar bir öncekinden daha iyi olacağı beklenen yeni bir popülasyon oluşturmak için kullanılır. Yeni popülasyon oluşturulması için seçilen çözümler uyumluluklarına göre seçilir. Çünkü uyumlu olanların daha iyi sonuçlar üretmesi olasıdır. Bu istenen çözüm sağlanıncaya kadar devam ettirilir (Kurt & Semetay, 2001). Bu işlemler yapılırken doğal seçim ilkesinden yararlanılarak popülasyonlardaki bireyler çaprazlama, mutasyon, seçim gibi işlemcilerden geçmektedir.

Genetik Algoritmaların aşamaları

Genetik Algoritmaların en basit formu olan standart olarak kabul edilmiş, ana çalışma prensibi Şekil 4.9'da verilen akış diyagramındaki gibidir.



Şekil 4.9 : Genetik sürecin yaşam döngüsü

Bu basit Genetik Algoritma şöyle çalışmaktadır:

- Genellikle rastgele seçilen bireylerle dolu olan ilk popülasyon (initial population) yaratılır.
- İlk popülasyondaki bireyler uygunluk fonksiyonuna göre değerlendirilirler.
- Popülasyonun bireyleri, yeniden üretim için bireylerin uygunluk değerleri temel alınarak seçilirler. Bu aşamada uygunluk orantısı, sıralanma, turnuvalı seçim gibi farklı seçim mekanizmaları da uygulanabilir.
- Yeni bireyler (çocuklar), ebeveyn bireylere genetik işlemciler uygulanarak elde edilir. Mevcut nesildeki bireyler yeniden üretim için seçilirler. Çaprazlama ile iki ayrı bireyin genetik kodları bir araya getirilir. Mutasyon ile yeni gen kodlarında kısmi değişiklikler yapılır. Karşılaşılan problemlerin özellikleri değiştikçe farklı çaprazlama ve mutasyon işlemcileri kullanılabilir.
- Elde edilen yeni bireyler uygunluk değerine göre değerlendirilirler. Yeni nesiller, mevcut popülasyona yeni, daha iyi özellikli bireylerin katılımıyla ve mevcutta bulunan bireylerin uygunluk değerini en az sağlayanlarının ölmeleriyle devam ettirilir.
- Sonlandırma kriteri elde edilene kadar, Genetik Algoritmanın bu yeniden üretim döngüsü mevcut nesildeki bireylerin yeniden üretim için seçilmesi aşamasından kendini tekrar eder. Bu tekrar sayısı sonuç çözümün elde edilebilmesi için sonlandırma kriterinin tatmin olmasına bağlı olduğu gibi özellikle belirlenmiş nesil sayısı kadar da olabilir (Holland, 1975; Goldberg, 1989; Davis, 1991).

Genetik Algoritma işlemcileri

Bir popülasyon oluşturulurken Genetik Algoritma yaklaşımında işlemciler kullanılmaktadır. Bunlardan sıklıkla kullanılan bazıları elitizm, seçim, çaprazlama ve mutasyon işlemcileridir.

Elitizm

Yeni bir popülasyon oluşturulurken, çaprazlama ve mutasyon operatörleri işlem yaparken popülasyondaki en iyi kromozomu büyük olasılıkla kaybetme şansı vardır. Elitizm, en iyi kromozomun (veya birkaç en iyi kromozomun) öncelikle yeni popülasyon için kopyalandığı metodun adıdır. Geri kalan kromozomlar klasik

yollarla elde edilir. Elitizm bulunan en iyi çözümün kaybedilmesini engellediği için Genetik Algoritmanın başarı performansını hızlıca yükseltebilir.

Seçim işlemcileri

Doğal evrimsel sürecin seçiminde sadece en uygun türler hayatta kalabilir, üreyebilir ve sonraki nesle onların genleri ile geçilebilir. GA da buna benzer bir seçim yaklaşımını kullanırlar. Seçim (selection) işlemcisi, kromozomların uygunluklarına bağlı olarak bir sonraki jenerasyonda yer alıp almayacaklarını belirleyen operatördür. Bu işlemci yeni topluluk içerisinde, uygunluğu yüksek bireylerin bulunmasını sağlamaktadır (Goldberg, 1989). Uygunluğu yüksek bireylerin hayatta kalma şansları daha fazladır. Doğal süreçten farklı bir şekilde, kromozom popülasyonunun boyutu bir nesilden diğer nesle değiştirilmemiş olarak aynı kalır.

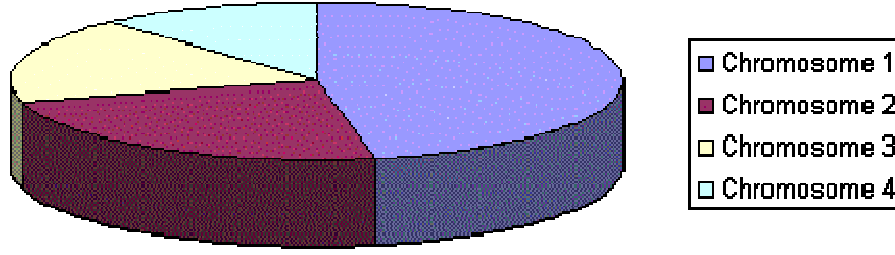
Başlangıç popülasyonu oluşturulduktan sonra yeni popülasyonun oluşturulabilmesi için seçim yöntemine karar verilmesi gerekmekte ve yüksek uyuma sahip bireylerin seçilme olasılığı daha fazla olmaktadır. Seçim işlemcileri ile tıpkı gerçek doğada olduğu gibi en iyilerin hayatta kalması sağlanmaktadır. Seçim işlemcisi ile seçilmiş olan bireyler çaprazlama havuzuna girmeye hak kazanırlarken seçilemeyen bireyler yok olmaktadır (Yurtçu, İçağa, 2006). Seçim işlemcisi için birçok yöntem geliştirilmiştir. Bunlardan en çok kullanılanlar rastgele seçim, rulet çemberi, sıralı seçim, turnuva seçimi ve sabit durum seçimidir.

• Rastgele seçim

Rastgele seçimde popülasyonda bulunan bütün kromozomların seçilme olasılıkları, uygunluk fonksiyonlarındaki değerlerine bakılmaksızın düzgün dağılıma (uniform) uygun olarak belirlenir. Çaprazlama işlemine maruz kalacak kromozomların seçimi herhangi bir kısıtlayıcı olmadan eş ağırlıklı olarak belirlenir (Haupt & Haupt, 2004).

• Rulet çemberi

Rulet çemberi seçiminde kromozomlar uygunluk fonksiyonun değer aralığına göre dairesel bir rulet etrafında toplanırlar. Uygunluk fonksiyonuna en yakın olan kromozomların yeri rulet çemberinde payı en fazla olandır (Şekil 4.10)



Şekil 4.10 : Rulet çemberi seçimi (Obitko, 1998).

Bu sisteminin mantığında bir adet bilye rulete fırlatılır ve bir aralıkta durması beklenir. Rulet üzerindeki uygunluk fonksiyonuna göre dilimlenmiş kromozomlardan payı en fazla olan en uygun kromozomların seçilmesi daha büyük olasılıklıdır. Bu algoritma şöyle açıklanabilir:

[Toplam] Popülasyondaki tüm kromozomların uygunluk fonksiyonlarının toplamı hesapla - toplam s .

[Seçim] $(0,s)$ aralığından rastgele bir sayı üret - r .

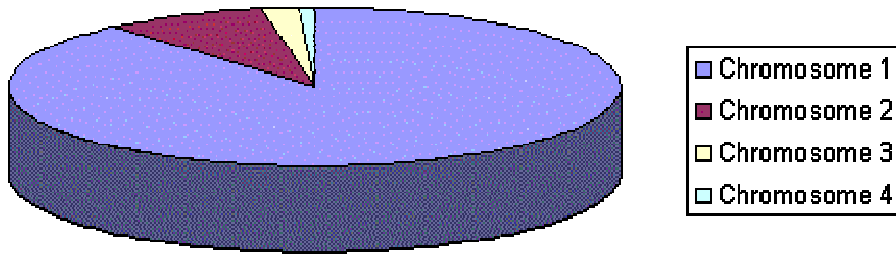
[Döngü] Popülasyon üzerinden gidip 0 'dan itibaren uygunluk değerlerinin toplamını al - toplam s . Eğer toplam s , r ' den büyük ise dur ve bulunan yerdeki kromozoma dön.

Aşama 1 her popülasyon için bir defa yapılmalıdır (Obitko, 1998).

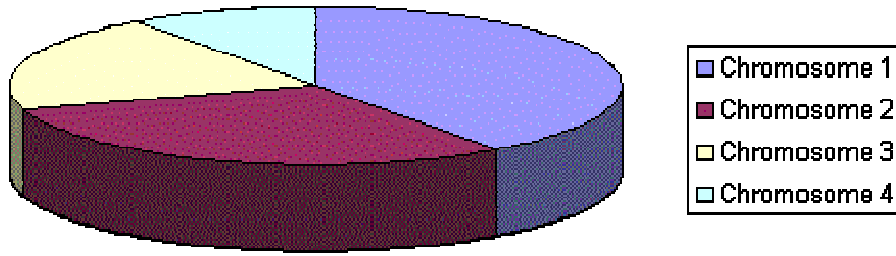
• Sıralı seçim

Bir önceki seçim sisteminde eğer uygunluk değerleri arasında çok büyük farklılıklar varsa problemler yaşanabilmektedir. Örnek olarak eğer en iyi kromozomun uygunluk değeri rulet çemberinin %90'ını kaplar ise diğer kromozomların seçilmek için çok az bir şansı kalır.

Sıralı seçimde ilk önce tüm nesil uygunluk değerinin verdiği sıraya göre sıralanır. En kötü 1 uygunluk değerini, ikinci en kötü olan 2 değerini alır ve en iyi olan ise N (popülasyondaki kromozom sayısı) değerini alır. Aşağıdaki grafiklerde uygunluk değerine göre yerleşen kromozomlar ile sıralı seçime göre sıralanan kromozomlar arasındaki fark görülmektedir (Şekil 4.11; Şekil 4.12).



Şekil 4.11 : Kromozomların uygunluk değeri sırasına göre dizilimi (sıralamadan önce) (Obitko, 1998).



Şekil 4.12 : Kromozomların uygunluk değeri sırasına göre dizilimi (sıralamadan sonra) (Obitko, 1998).

Bu dizilimden sonra bütün kromozomların seçilmek için şansı olacaktır. Bu metot en iyi kromozomlar arasında çok fazla farklılık olmayacağı için daha yavaş yakınsamaya neden olabilir (Obitko, 1998).

• Turnuva seçimi

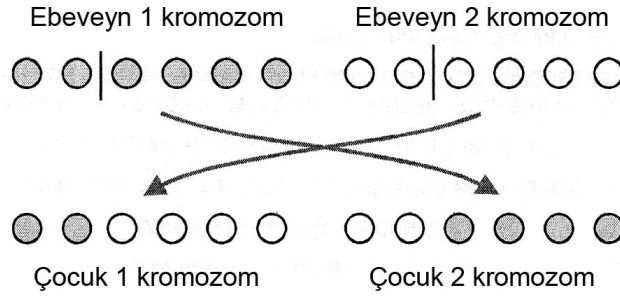
Turnuva seçim yönteminde yerine koyarak ya da koymadan rastgele t adet birey seçilir ve bu büyüklüğe turnuva genişliği adı verilir. Bu gruptaki en iyi birey, yeni popülasyona kopyalanır. Bu işlem kullanıcı tarafından önceden kararlaştırılan çevrim sayısı kadar tekrarlanır (Yurtçu, İçağa, 2006).

• Sabit durum seçimi

Bu yöntem bir ebeveyn seçme mantığı üzerinden yürümez. Bu seçim yönteminin ana fikri kromozomların büyük kısmının gelecek nesle aktarılmasıdır. Burada GA şöyle çalışmaktadır. Her nesilden bazı kromozomlar (iyi – yüksek uygunluk değerine sahip olanlar) yeni çocukların yaratılması için seçilirler. Sonra bazı kromozomlar (kötü – düşük uygunluk değerine sahip olanlar) yeni gelen çocuklara yer vererek ortadan kalkarlar. Geri kalan popülasyon aynen yeni nesle aktarılır.

Çaprazlama

Çaprazlama (Crossover) Genetik Algoritmaların devamlılığının sağlanabilmesi için uygulanan en önemli işlemcilerden biridir. Bu işlemci ile arama uzayındaki iki kromozom alınıp, birbirleri arasında rastgele gen alışverişinde bulunurlar ve böylece her ebeveyninden belirli genotipleri alan farklı genotiplere sahip çocuk kromozomlar oluşturulur (Şekil 4.13).

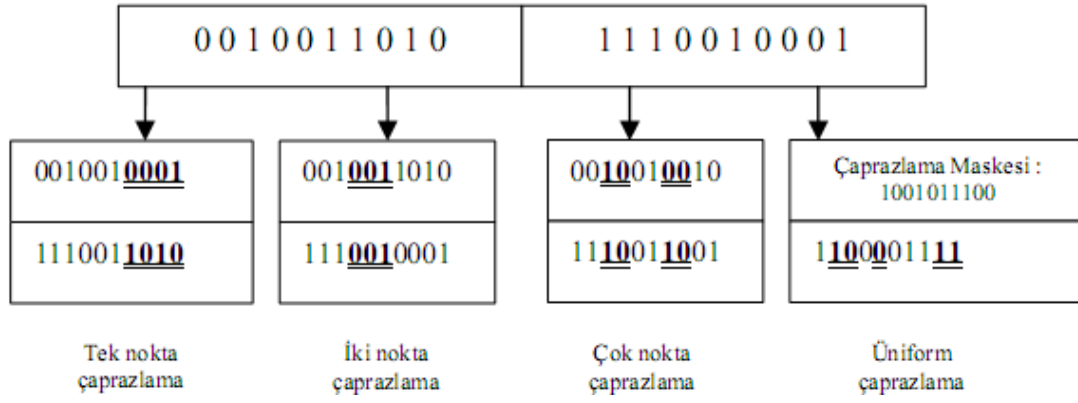


Şekil 4.13 : Basit çaprazlama operatörü (Bentley, 1999).

Oluşan yeni birey, ebeveynlerinin bazı özelliklerini almış ve bir bakıma ikisinin kopyası olmuştur. Çaprazlama işlemi başka şekillerde de yapılabilir. Birden fazla çaprazlama noktası gibi daha iyi performans almak amacıyla değişik çaprazlamalar kullanılabilir (Kurt & Semetay, 2001). Popülasyonda ne oranda bir çaprazlamaya izin verildiği çaprazlama oranı (pc) ile gösterilir.

Karşılaşılan problemlere bağlı olarak, kullanıcı tarafından seçilen 4 farklı çaprazlama operatörü bulunmaktadır (Şekil 4.14).

- Tek nokta çaprazlama
- İki nokta çaprazlama
- Çok nokta çaprazlama
- Uniform çaprazlama



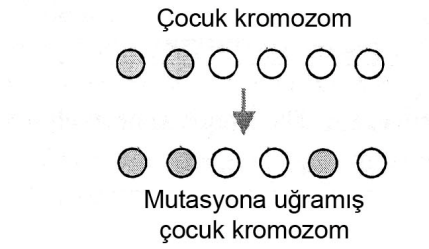
Şekil 4.14 : Çaprazlama yöntemleri ve etkileri, çaprazlama yapılanların altları çizilmiştir (Bolat, Erol ve İmrak, 2004).

Tek nokta çaprazlama, Genetik Algoritmanın kullandığı en basit çaprazlamadır. Rastgele seçilen kromozom çiftine çaprazlama uygulanır. Tek nokta çaprazlama işlemi için kromozomda çaprazlama yapılacak bölge kullanıcı tarafından rastgele seçilebilir. Oluşan yeni birey ebeveynlerin bazı özelliklerini alarak her ikisinin kopyası olacaktır (Eksin & Erol, 2001).

İki nokta çaprazlamada, iki nokta arasında kalan alt dizilerin değiştirilmesiyle iki yeni birey elde edilir. Çok nokta çaprazlama yöntemi, iki nokta çaprazlamanın gelişmiş bir halidir. Kromozomlar daha fazla parçalara ayrılır ve bir atlanarak elde edilen çiftler arasında değiştirilerek yeni bireyler elde edilir. Üniform çaprazlamada rastgele olarak çaprazlama maskesi oluşturulur. Birinci ve ikinci kromozoma karşılık gelen genin kopyalanmasıyla oluşturulur. Çaprazlama maskesinde bir o genin birinci kromozomdan, sıfır ise o genin ikinci kromozomda kopyalanacağı anlamına gelmektedir (Bolat, Erol ve İmrak, 2004).

Mutasyon

Çaprazlama operatörleri işlemini tamamladıktan sonra mutasyon (mutation) işlemine başlanır. Mutasyonda kromozomların genlerinin bir ya da birkaçının yerleri rastgele değiştirilerek yeni kromozomlar elde edilir (Şekil 4.15). İkili kodlama sisteminde bu operatör işlemi gen değeri 1 ise 0, 0 ise 1 olarak değiştirilerek gerçekleştirilir. Mutasyon işlemcisi, çözüm uzayında bir daha elde edilemeyecek sonuçların kaybına karşı koruma sağlamaktadır (Goldberg, 1989). Böylece yeni oluşturulan bireylerin, nesildeki eski bireylere benzemesi önlenir, popülasyondaki çeşitlilik azalmaz ve aynı zamanda evrimleşme süreci hızlanır. Mutasyon sırasında kromozomlarda bulunan gen sayısı değişmez. Mutasyon oranı GA'nın frekansını da etkileyen bir değerdir.

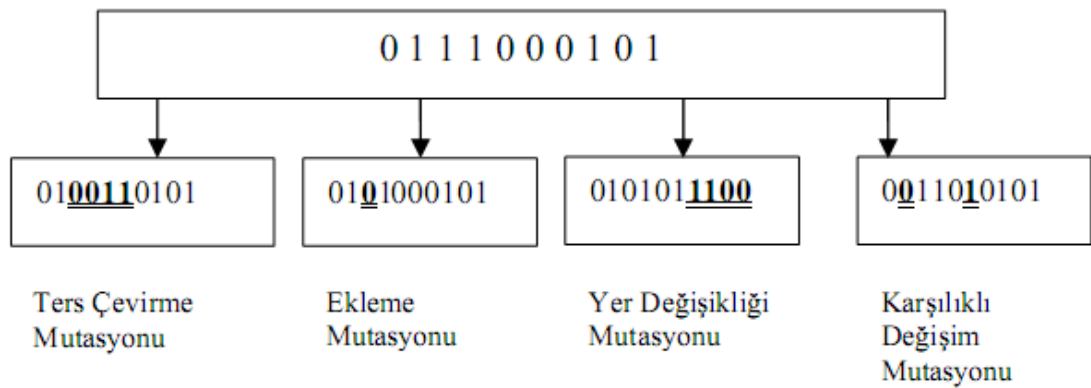


Şekil 4.15 : Basit mutasyon operatörü (Bentley, 1999).

Örnek olarak ikili bir kodlamanın kullanıldığı bir dizide mutasyon işlemcisi ile rastgele olarak seçilen eleman değeri 1 ise 0 veya 0 ise 1 olarak değiştirilerek yeni bir dizi elde edilir. Mutasyon operatörü olarak, ele alınan problemin yapısına göre en uygun olan aşağıdakilerden biri seçilir.

- Ters çevirme,
- Ekleme,
- Yer değişikliği,
- Karşılıklı Değişim.

Mutasyon operatörlerinin uygulamaları Şekil 4.16’da görülmektedir. Şekilde altı çizili olarak verilen eleman değerleri mutasyona uğramış elemanları göstermektedir. Ters çevirmede, kromozomdan rastgele iki pozisyon seçilir ve iki ucu arasında ters çevrilir. Eklemede ise rastgele bir parka seçilir ve rastgele bir yere yerleştirilir. Yer değişikliği mutasyonunda, rastgele bir alt dizi seçilir ve rastgele bir yere yerleştirilir. Karşılıklı değişim mutasyonunda, rastgele seçilen iki genin yerleri değiştirilir (Bolat, Erol ve İmrak, 2004).



Şekil 4.16 : Mutasyon yöntemleri ve etkileri (Bolat, Erol ve İmrak, 2004).

Genetik Algoritmaların işlemcilerle çalışması üzerine basit bir örnek:

Aşağıda rastgele seçilmiş 8 bitlik 4 adet birey görülmektedir. Uygunluk kriteri her dizinin barındırdığı 1 adedi olsun (a bireyinde 2 adet 1 bulunmaktadır dolayısıyla uyumluluğu 2'dir);

a 00000110 2

b 11101110 6

c 00100000 1

d 00110100 3

Bu durumda aradığımız değer 8 uyumluluğa sahip bir değer olsun. Bu 4 adet dizi içinden ikişer çift seçerek sonraki kuşak oluşturulacak ve bunlara çaprazlama ve mutasyon uygulanacaktır. b,d ile b,c çiftleri seçilsin. Çaprazlama bu çiftler içinde çaprazlama olasılığı oranında gerçekleşecektir. b,d çiftinde çaprazlama olduğunu varsayalım. Bu durumda e=10110100 ve f=01101110 şeklinde yeni bireyler oluşacaktır. Burada çaprazlama ilk bit üzerinden gerçekleşmiştir. Diğer çift üzerinde çaprazlama yapılmasın. Mutasyon ise mutasyon olasılığı oranında etki edecektir. Burada e ile b bireylerine etkimiştir (e'ye 6. bit, b'ye ilk bit üzerinden). Mutasyona uğrayan bitler değişir ve son durum aşağıdaki gibi olur;

e* 10110000 3

f 01101110 5

c 00100000 1

b* 01101110 5

Görüldüğü gibi en yüksek uyumluluğa sahip olan b bireyinin kaybolmasına karşılık ortalama uyumluluk artmıştır. Bu işlem birkaç kez tekrar edildiğinde GA mükemmel diziyi yani 11111111 dizisini bulacaktır (Kurt & Semetay, 2001).

4.1.4.3 Genetik Algoritma parametreleri

Genetik Algoritmaların etkinliği kullanılan parametrelere bağlı olarak değişir (Chen ve diğ., 1995). Değiştirilen parametreler ile GA'nın kalitesi de artar ve daha sağlıklı sonuçlar elde edilir. Bu parametrelerden bazıları şunlardır:

- Populasyon Büyüklüğü,

- Çaprazlama Olasılığı,
- Mutasyon Olasılığı,
- Kuşak Farkı.

Popülasyon büyüklüğü GA kullanıcısı tarafından verilen en önemli kararlardan birisidir. Bu değer çok küçük olduğunda, Genetik Algoritma yerel bir optimuma takılabilmektedir. Popülasyonun çok büyük olması ise çözüme ulaşma zamanını artırmaktadır. Bu konuda Goldberg 1985'te, yalnızca kromozom uzunluğuna bağlı bir popülasyon büyüklüğü hesaplama yöntemi önermiştir (Emel & Taşkın, 2002). Daha sonra yapılan test fonksiyonları göstermiştir ki, popülasyon büyüklüğü 20-30 birey arasında sınırlı kalırsa elde edilen algoritma sonuçları yeterince iyi olmaktadır.

Çaprazlama olasılığı ne kadar sıklıkla çaprazlamanın yapılacağını gösterir. Çaprazlamanın amacı, mevcut iyi kromozomların özelliklerini birleştirerek daha uygun kromozomlar yaratmaktır. Kromozom çiftleri $P(c)$ olasılığı ile çaprazlamaya uğramak üzere seçilirler. Çaprazlamanın artması, yapı bloklarının artmasına neden olmakta fakat aynı zamanda bazı iyi kromozomların da bozulma olasılığını artırmaktadır (Emel & Taşkın, 2002).

Eğer herhangi bir çaprazlama yapılmazsa çocuk kromozomlar ebeveynlerinin birebir kopyası olurlar. Çaprazlama yapılırsa, elde edilen kromozomlar atalarının birer parçası olurlar. Eğer çaprazlama olasılığı %100 ise çocuk kromozomlar tamamen çaprazlama ile elde edilirler, bununla birlikte olasılık %0 olursa bütün yeni nesil, eski popülasyondan kopyalanmış kromozomlara sahip olur (Fakat bu yeni neslin, eskisi ile birebir aynı olduğunu göstermez). Çaprazlama yeni kromozomların, eski kromozomların iyi taraflarını alıp popülasyonu daha iyi hale getireceği düşünülerek yapılır. Ancak eski popülasyonun bazı parçalarının yeni nesle aktarılması çeşitliliği artırması açısından da iyidir.

Mutasyon olasılığı, kromozom parçalarının hangi sıklıkla mutasyon geçireceğini bildirir. Eğer mutasyon olmazsa, çocuk kromozom çaprazlamadan veya kopyalamadan sonra herhangi bir değişiklik yapılmadan üretilir. Mutasyon olursa, kromozomun bazı bölümleri değişir. Eğer mutasyon olasılığı %100 ise bütün kromozom yapısı değişir, %0 olursa hiçbir değişikliğe uğramaz. Mutasyon GA'nın yerel ekstremelere düşmesini engellemek amacıyla yapılır, fakat GA'nın rastgele arama çalışmasına dönüşmemesi için çok sık yapılmamalıdır.

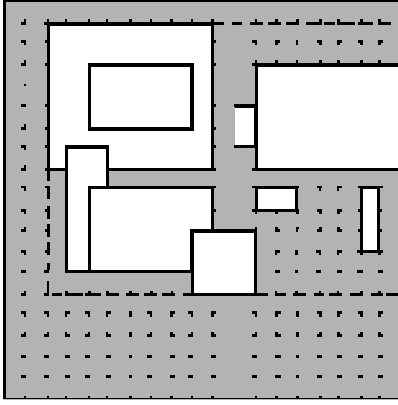
Mutasyonun amacı popülasyondaki genetik çeşitliliği korumaktır. Mutasyon $P(m)$ olasılığı ile bir kromozomdaki her bitte meydana gelebilir. Eğer mutasyon olasılığı artarsa, genetik arama rastsal bir aramaya dönüşür. Fakat bu aynı zamanda kayıp genetik malzemeyi tekrar bulmada yardımcı olmaktadır (Emel & Taşkın, 2002).

Bir popülasyonda yeni bireyler uygunluk değerine göre veya rastgele olarak seçilebilirler. Yeni bireylerin tamamen rastgele seçilmesi durumunda GA'nın çözüm uzayında yakınsaması zorlaşabilir. Tüm bireyler uygunluk değerine göre seçildiğinde ise oluşturulan yeni nesil içerisinde lokal yakınsamalar olabilir. Bu tür sorunlar her kuşakta belirli bir oran içerisinde bazı kromozomların uygunluk değerine göre bazılarının ise rastgele seçilmesi ile aşılar. Bu oran Kuşak farkı (Generation Gap) ile ifade edilir. Kuşak farkı algoritmadaki operatörler için kaç adet kromozomun seçildiğini göstermektedir; bu oran %100 olduğunda kromozomların tamamının yer değiştirdiği anlamına gelmektedir.

4.1.4.4 Genetik Algoritma ile örnek problem çözümü

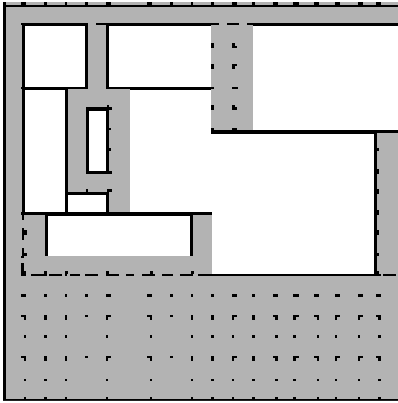
Genetik Algoritmaların çalışma mantığının açıklanması amacıyla dikdörtgen yerleştirme probleminin çözümüne yer verilecektir.

Ele alınan problem farklı boyutlardaki verilmiş dikdörtgenlerin tümünü, alanı mümkün olan en küçük bir dikdörtgen içine yerleştirmektir. Bu konuşlandırma sırasında hiç bir dikdörtgenin bir diğeri ile kısmen de olsa örtüşmemesi ve her dikdörtgenin bir kenarının düşey doğrultuda olması gerekmektedir. Dolayısı ile bir dikdörtgen ya verildiği üzere ya da 90° çevrilmiş (devrik) olarak yerleştirebilmektedir. Dikdörtgenlerin boyutları tamsayı olarak verilmektedir. Aşağıdaki şekilde verildiği varsayılan bir küme dikdörtgenin rastgele bir yerleşimi görülmektedir (Şekil 4.17). Bu yerleşim $13 \times 17 = 221$ birim² lik dikdörtgensel bir alan içinde kalmakla beraber bazı dikdörtgenlerin örtüşmesinden ötürü geçerli bir çözüm değildir.



Şekil 4.17 : Dikdörtgenlerin rastgele yerleşimi (Üçoluk, 2002).

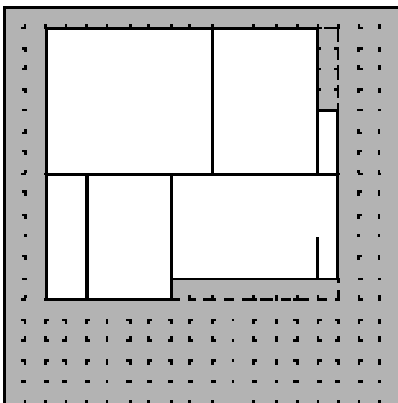
Şekil 4.18’de geçerli bir yerleşim görünmektedir:



Şekil 4.18 : Dikdörtgenlerin geçerli yerleşimi (Üçoluk, 2002).

Bu yerleşim $12 \times 17 = 204$ birim² lik toplam alana sahip geçerli bir yerleşimdir.

Ancak Şekil 4.19’deki yerleşim probleme daha iyi bir çözüm önermektedir:



Şekil 4.19 : Dikdörtgenlerin geçerli yerleşimi (Üçoluk, 2002).

Bu yerleşim $13 \times 14 = 182$ birim² lik bir alana sahiptir (Üçoluk, 2002). Bu problem göstermektedir ki GA ile bir problemin çözümü, olasılıklardan en iyi olanlara yakınsar.

Adım adım bir problemin GA yöntemi ile çözülmesi için neler yapılması gerektiği yukarıda oluşturulan örnek üzerinden irdelenecektir.

. Problem çözüm uzayı $\mapsto$ İkili sayı dizgileri uzayı

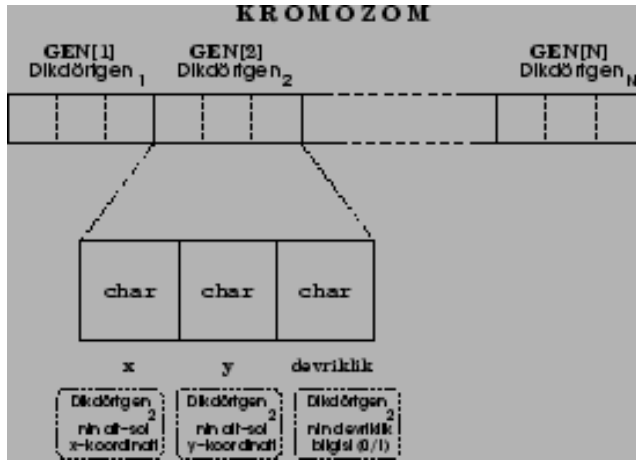
betimlemesi yapılmalıdır. Çoğu uygulamada ikili sayı dizgisinin (kod) boyu çözüm işlemi boyunca sabit tutulmaktadır.

Örnek problemde tüm bellekten yer almaların dinamik gerçekleştirildiği, i nci dizi öğesinin i nci dikdörtgenin sol-alt köşesinin yerleşim koordinatları ve dikdörtgenin devriklik bilgisini (örneğin 0:devrik değil, 1:devrik) tutan 1-boyutlu bir dizi gösterimi her bir dizi öğesi (gen)

```
struct gene
```

```
{ char x, y, orientation; };
```

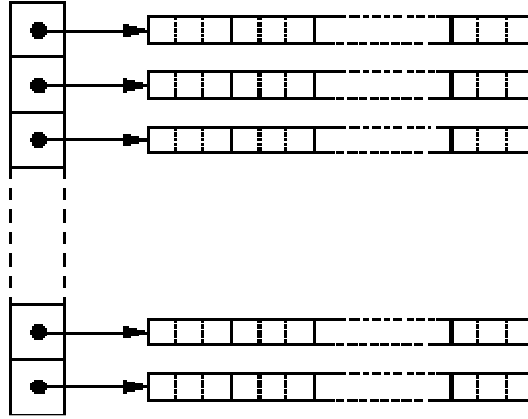
biçiminde bir kayıt olacaktır. Diziye bir bütün olarak bakıldığında, problem bir çözüm adayını kodlayabilen bir ikili sayı dizgisi olarak algılanmaktadır (Şekil 4.20).



Şekil 4.20 : Kromozomları oluşturan genlerin kodlanması (Üçoluk, 2002).

- Belirlenmiş türden dizgileri tutabilecek bir havuz oluşturulur. Bu, genellikle elemanları dizgiler olan bir dizi değişkeni olarak gerçekleştirilir. Yeni kuşak oluşturulduktan sonra kısa bir süre için eski kuşağa da ihtiyaç olacaktır. Dolayısı ile havuz oluşturulurken yeni kuşak kromozomların konacağı ikinci bir havuz oluşturmak iyi bir tekniktir (Şekil 4.21).

HAVUZ



Şekil 4.21 : Belirlenmiş dizgeler havuzda tutulur (Üçoluk, 2002).

- Düzgün bir rastgelelikle dizgilerin ilk değerleri oluşturulur.

. **Uygunluk** : İkili sayı dizgileri uzayı $\mapsto \mathcal{R}$

olacak biçimde bir **Uygunluk** fonksiyonu tanımlanır. **Uygunluk(s)**, **s** dizgisinin gerçek bir problemin ne denli başarılı bir çözümünü kodladığının ölçütü olacaktır.

Örnek problemde böyle bir ölçütü tanımlayabilmek için bir çözümün iyilik derecesine ilişkin nitel bileşenler bulunması mantıklı olacaktır. Bir kromozoma ilişkin aşağıdaki bileşenler önerilebilir:

Sayı_{saha dışı}: Tüm köşeleri yerleştirme ızgarasının içinde olmayan dikdörtgenlerin sayısı.

Yüzölçüm_{KapsayanDD}: Saha dışı olmayan tüm dikdörtgenleri içine alan en küçük dikdörtgen alanın yüzölçümü.

Yüzölçüm_{Örtüşen}: Saha dışı olmayan dikdörtgenlerin örtüşen alanlarının toplamı.

Şüphesiz iyi denilebilecek bir çözümde **Sayı_{saha dışı}** ve

Yüzölçüm_{Örtüşen} değerlerinin sıfır, **Yüzölçüm_{KapsayanDD}** değerinin de küçük olması gerekir.

Ancak GA uygulamalarında önemli bir nokta bu fonksiyonun sürekliliğinin elden geldiğince gözetilmesidir. Sürekli bir *Uygunluk* fonksiyonu GA'ya çözümlerini iyileştirmesinde yol gösterici olacaktır.

Bu bileşenleri tek bir toplam altında birleştirip her çözüm adayına ilişkin bir uygunluk değeri oluşturulabilir:

$$C_1 \times \text{Sayı}_{\text{saha dışı}} + C_2 \times \text{Yüzölçüm}_{K\text{apsayan}DD} + C_3 \times \text{Yüzölçüm}_{\text{Örtüşen}}$$

C_1 , C_2 ve C_3 sabitlerinin pozitif olması durumunda yukarıda hesaplanan değer in sıfıra yakınlığı 'en uygunluk' anlamında olacaktır. Pratikte C_1 ve C_3 değerlerinin göreceli (C_2 ye göre) büyük seçilmesi saha dışılığı ve örtüşmeyi engelleyecektir.

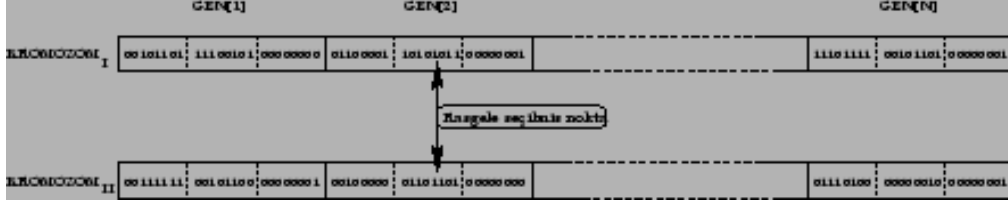
- Havuzdaki tüm bireyler (kromozomlar) *Uygunluk()* fonksiyonu aracılığı ile değerlendirilip sonuçları saklanır. Genellikle bu, elemanları kayan noktalı sayılar olan, havuz değişkenine koşut, bir dizi değişkeninde gerçekleştirilir.

- Havuzdan rastgele iki dizgi seçilip, bunlar yine rastgele belirlenen bir noktadan ikiye kesilir. Oluşan parçalar değiş tokuş edilerek yeni iki dizgi elde edilir, bu yeni dizgiler yukarıda açıklandığı üzere 'yeni' havuzda saklanır. Bu işlem havuzda çaprazlanmamış dizgi kalmayıncaya kadar sürdürülür. Bazı GA uygulamalarında bir kromozomun çaprazlamada kullanılması uygunluk değeri ile koşut bir olasılık ile belirlenir. Yani uygunluk değeri yüksek bireyler daha fazla çaprazlamada yer alırken, uygunluk değeri düşük olanlar en fazla bir (bazen hiç) olmak üzere çaprazlanırlar.

Örneğe ilişkin, çaprazlama işlemine daha yakından bakalım. Şekil 4.22’de iki kromozomun çaprazlanmak üzere eşleştirildiğini varsayalım:

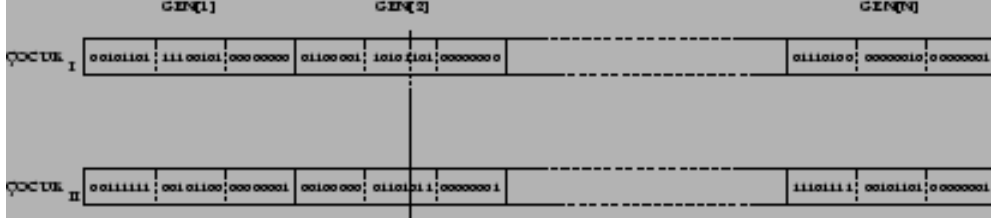
	GEN[1]			GEN[2]			GEN[N]		
KROMOZOM I	45	229	0	97	171	1	235	45	1
KROMOZOM II	63	44	1	32	109	0	116	2	1

Şekil 4.22 : Çaprazlanmak üzere eşleştirilmiş iki kromozom (Üçoluk, 2002).



Şekil 4.23 : İki kromozomun çaprazlanmasında rastgele seçilmiş çaprazlama noktası (Üçoluk, 2002).

Daha yakından (bit düzeyinde) bakar ve rastgele seçilmiş çaprazlama noktası göz önüne alınırsa, Şekil 4.23’de görüldüğü üzere çaprazlama noktası büyük bir olasılıkla bazı ara bitlere denk düşecektir. Uygulama yazılımının oluşturulmasında bu durumun göz önünde bulundurulması ve çocukların kromozomların sağlıklı üretilmesinin önlemlerinin alınması gereklidir. Aşağıdaki Şekil 4.24’de çaprazlama sonrası oluşan yeni kromozomları (çocukları) görülmektedir.



Şekil 4.24 : Çaprazlama sonrası oluşan yeni kromozomlar (Üçoluk, 2002).

Bit diziliminin karşılık geldiği sayılar ise Şekil 4.25’de görülmektedir.

	GEN[1]			GEN[2]			GEN[N]		
ÇOCUK I	45	229	0	97	173	0	116	2	1
ÇOCUK II	63	44	1	32	107	1	235	45	1

Şekil 4.25 : Çaprazlama sonrası oluşan yeni kromozomların bit dizilimine göre aldığı değerler (Üçoluk, 2002).

Bu işlemin sonucunda, ata kromozomların bulunduğu havuza ek olarak bir de çocuk kromozomları içeren yeni bir havuz oluşturulmuş olmaktadır.

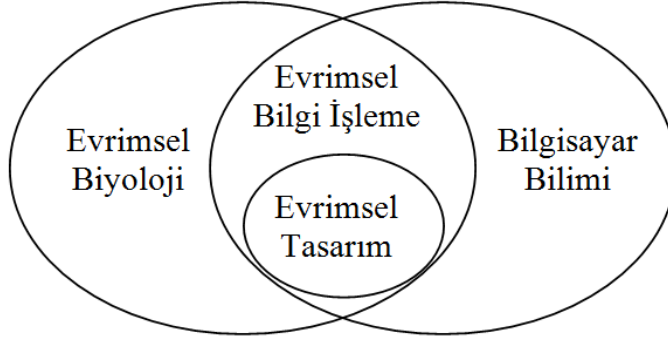
- Mutasyon oranı ile yeni havuzdaki bireylerden bazıları rastgele seçilip rastgele bir bit'i değiştirilebilir.
- Yeni havuzdaki tüm dizgiler *Uygunluk()* fonksiyonu aracılığı ile değerlendirilir. Yeni havuzdaki bireylerin (çocukların) sayısının eski havuzdaki (ata) bireylerin sayısı ile aynı olmalıdır.

- Yeni oluşturulan havuzdaki bireylerden hangilerinin yaşatılacağına ve hangilerinin öldürüleceğine karar verilir. Bunun için sık kullanılan bir yol, seçkinlik gözetmektir. Seçkinlik 'en uygun çözüm olma' anlamındadır. Eski havuzdaki en seçkinler (yani uygunluk değerleri en yüksek olanlar) arasından n tanesi bir kuşak daha yaşayabileceklerdir. Bu seçkin n birey dışındaki tüm eski havuz bireyleri öldürülüp yerlerine yeni havuzdakiler (en kötü n tanesi hariç olmak üzere) alınır. Uygulamalarda, n in toplam birey sayısına oranı %5 dolayındadır.
- İşlemi durdurma ölçütü sağlanmamışsa işleme devam edilir. İşlemi durdurma çoğunlukla belirli bir kalitede çözüme varılmış olması, belirli sayıda kuşak oluşturulmuş olması veya uzun süreli olarak havuzda çözüm iyiliği bakımından bir değişiklik gözlenmemesi gibi bir ölçüte bağlanmaktadır (Üçoluk, 2002).

4.2 Evrimsel Tasarım

Şimdiye kadar Evrimsel Yaklaşımlar kapsamında bir arama uzayında iyi sonuçlar elde etmek için bilgisayarların nasıl kullanıldığı anlatıldı. Bu tür evrimsel algoritmalar tarafından kullanılan birbirine paralel arama mekanizmaları olan Evrim Stratejileri, Evrimsel Programlama, Genetik Programlama ve Genetik Algoritmalar açıklandı. Bu evrimsel Algoritmaların mekanizmalarının aynı temel mantığa oturduğu ve farklı yollarla geliştiği gözlemlendi. Evrimleşme için bilgisayarların nasıl kullanıldığı ve bilgisayarların evrimsel tasarımı nasıl gerçekleştirdiği ise bu bölümde açıklanacaktır.

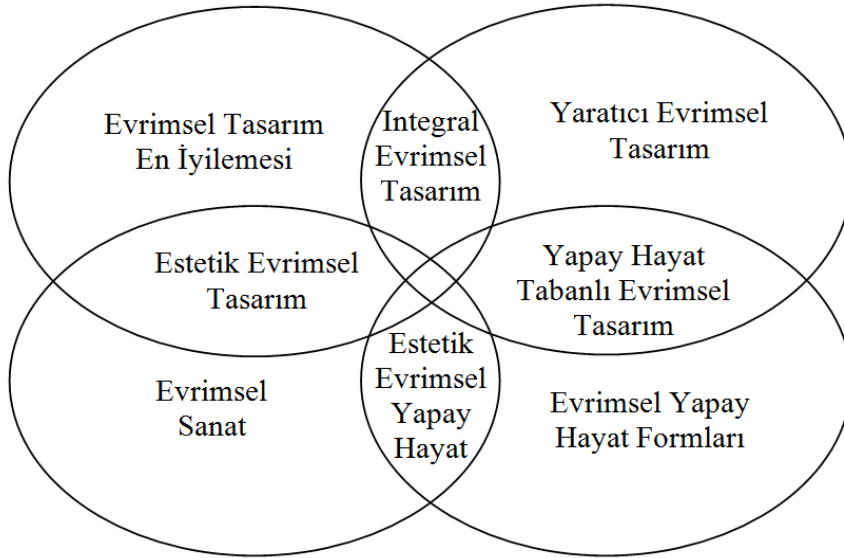
Evrimsel Tasarımın bilgisayar bilimine, tasarıma ve evrimsel biyolojiye dayanan kökleri vardır. Aynı zamanda CAD ve analiz yazılımlarının bir uzantısı olan ve doğal evrimin fikirleriyle desteklenen evrimsel tasarım, evrimsel bilgi işlemenin bir dalıdır (Şekil 4.26).



Şekil 4.26 : Evrimsel bilgi işleme ve evrimsel tasarım, bilgisayar bilimleri ve evrimsel biyolojinin bir araya gelmesiyle ilişkilidir (Bentley, 1999).

Bilgisayarlı Evrimsel Tasarım da kendi içerisinde farklı görüşlere ayrılmaktadır (Şekil 4.27). Bunların temel olanları şöyledir:

- Evrimsel Tasarım En İyilemesi (Optimizasyonu)
- Yaratıcı Evrimsel Tasarım
- Evrimsel Sanat
- Evrimsel Yapay Hayat Formları



Şekil 4.27 : Bilgisayarlı evrimsel tasarım görüşleri (Bentley, 1999).

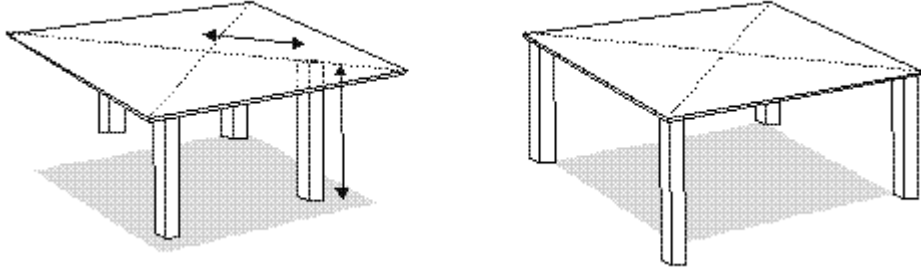
4.2.1 Evrimsel tasarım en iyilemesi

Evrimsel bilgi işlemenin, hali hazırda bulunan tasarımların en iyilemesinde kullanılması evrimsel tasarımın bir türüdür. Geçen on yıllarda, motor volanı ve uçak geometrisi gibi birçok farklı türlerdeki mühendislik tasarımları başarıyla geliştirilip optimize edilmiştir (Bentley, 1999).

Gerçek olan şudur ki, bu tür evrimsel tasarımlar üretken (generative) veya yaratıcı (creative) tasarım olarak adlandırılmazlar. Evrimsel Tasarım En İyilemesi konusunda çalışanlar var olan bir tasarımı ele alıp geliştirilmesi gereken yerleri belirleyip, bu belirlenen kısımları parametrelendirirler. Bu parametreler daha sonra genler olarak ve değerler olarak kodlanır ve evrimsel arama algoritmaları tarafından evrimleştirilirler. Tasarımlarda genellikle her tasarım için bir uygunluk değeri türetmek amacıyla kullanılan analiz yazılımı, sistemdeki arabirim tarafından değerlendirilir. Örneğin, Goodman motor volanlarını Genetik Algoritma (GA) kullanarak optimize ettiğini açıklamıştır. Motor volanlarının enine kesitleri parametrelendirilmiştir (yükseklik parametrelerinin bir toplamı halinde) ve geliştirilmiştir (Bentley, 1999).

Evrimsel Tasarım En İyilemesi üzerine masa örneği:

Evrimsel Tasarımın bu türünü göstermek için dört ayaklı masa tasarımı örneği burada açıklanacaktır. Evrimsel tasarım en iyilemesi için tipik bir yaklaşım, tasarımın parçalarını, ayakların yeri ve uzunluklarını parametrelendirmek ve bir evrimsel algoritma kullanılarak bu parametrelerin değerlerini bazı kriterlere göre optimize etmek olacaktır. Örnek olarak bu kriterlerden biri masanın sabitliğini maksimum değerlere çıkarmak olabilir. Aşağıdaki şekilde de görüldüğü üzere fenotipler sekiz parametreden ve genotipler 64 bit değerinden oluşmaktadır (Şekil 4.28).



Fenotip:

Masayı oluşturan tabla ve dört ayağı şöyle tanımlanmıştır:

1. Ayak Uzunluğu, 1. Ayağın masanın merkezine mesafesi
2. Ayak Uzunluğu, 2. Ayağın masanın merkezine mesafesi
3. Ayak Uzunluğu, 3. Ayağın masanın merkezine mesafesi
4. Ayak Uzunluğu, 4. Ayağın masanın merkezine mesafesi

Genotip:

11010110	10101101	10101110	10011010	01101010	10001010	11110010
00101110						

Uzunluk 1 Mesafe 1 Uzunluk 2 Mesafe 2 Uzunluk 3 Mesafe 3 Uzunluk 4 Mesafe 4

Şekil 4.28 : Bir masanın evrimsel tasarım en iyilemesi (Bentley, 1999).

Bu basit örnekte, en iyileme çözümü açıkça dört bacağı aynı uzunlukta olan bir masa tasarımı yapılmıştır ve masanın ayakları masa tablasının dört köşesine gelecek şekilde sabitlenmiştir. Örnekten de anlaşılacağı üzere Evrimsel Tasarım En İyilemesi var olan bir tasarımda işlevleri bakımından optimum (veya en azından işlev olarak iyi) çözüm formunu bulmaktadır. Fakat masa tablasını geniş bir tabanı olan tek bir kaide üzerine oturtmak gibi bir yaratıcılıkla tasarım anlayışını değiştirebilme yeteneğine sahip değildir.

4.2.2 Yaratıcı evrimsel tasarım

Bilgisayar tarafından üretilen herhangi bir şeyin yaratıcı olarak tabir edilmesi bir tartışma konusudur. Burada yaratıcı kelimesinin evrimsel tasarımda kullanımı açıklanacaktır.

Bir birey yaratıcılığı tasarlarken gösterebilir ve bir tasarım yaratıcı olmak üzere amaçlanan özelliklere sahip olabilir. Gero yaratıcı tanımının resmi olan kısmı üzerinde durur ve bir bilgisayar yaratıcı tasarım yaparken bağımsız tasarım uzaylarını bulmanın yanı sıra bir dizi olası tasarım durum uzayı keşfettiğini söyler. Bir başka deyişle Gero, karar değişkenlerinin çeşitlerinin geliştirilmesi ve bu değişkenlerin değerlerinin de evrimleşmesi ile bir bilgisayarın yaratıcı olduğu görüşünü savunmaktadır (Gero, 1996). Benzer bir şekilde, Boden kitabında yaratıcılığın bir sunumun sınırlarının ötesinde, farklı, alışılmamış ve o sunumun bile içerisinde gösterilemeyeceğini savunmaktadır. Buna rağmen bilgisayarların bu tür bir yaratıcı kapasiteye sahip olduğunu düşünmemektedir (Boden, 1990).

Bütün Yaratıcı Evrimsel Tasarım sistemlerin ortak, ana özelliği küçük bir noktadan veya sıfırdan başlayarak yeni formlar üretme kabiliyetleri olmasıdır (rastgele üretilen ilk popülasyonlar gibi) ve işlevsel performans kriterleri ile çok iyi yönlendirilmeleridir. Bunu başarabilmek için, bu tür sistemlerde evrimleşme sırasında karar değişkenlerinin sayılarını çeşitlendirmek gerekmektedir. Genellikle beklenmedik sürpriz çözümler veya diğerlerinden daha kaliteli alışılmamış çözümler üretirler. Sonuç olarak denilebilir ki bu tür sistemler gerçekten ‘yaratıcı olarak tasarlamak’ tadır.

Amanda Levete Architects tarafından tasarlanan bir süper yüzey formu (supersurface) 2009 Londra Tasarım Festivalinde sergilenmiştir. Bu morfo-genetik bilgisayar yazılımı aracılığıyla elde edilmiş bir Yaratıcı Evrimsel Tasarım örneğidir ve gelişen bu form evrimleşme halindeyken alınan bir anlık veri ile fabrika üretimi yapılmıştır (Şekil 4.29).

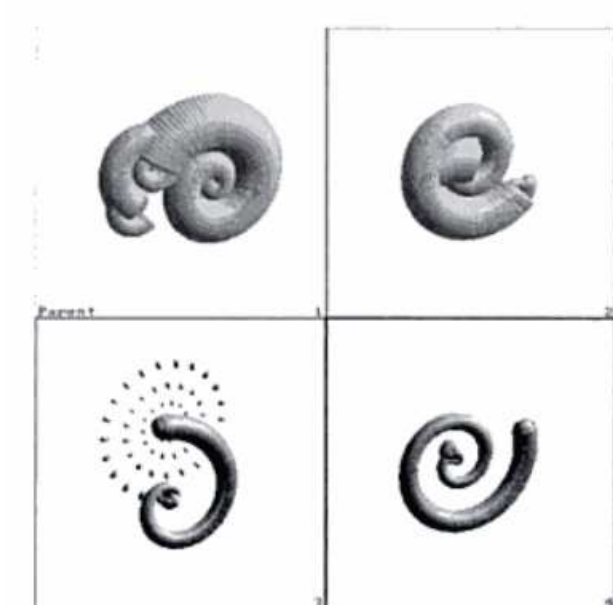


Şekil 4.29 : Yaratıcı evrimsel tasarım örneği süper yüzey
(<http://www.amandalevetearchitects.com/news/>).

4.2.3 Evrimsel sanat

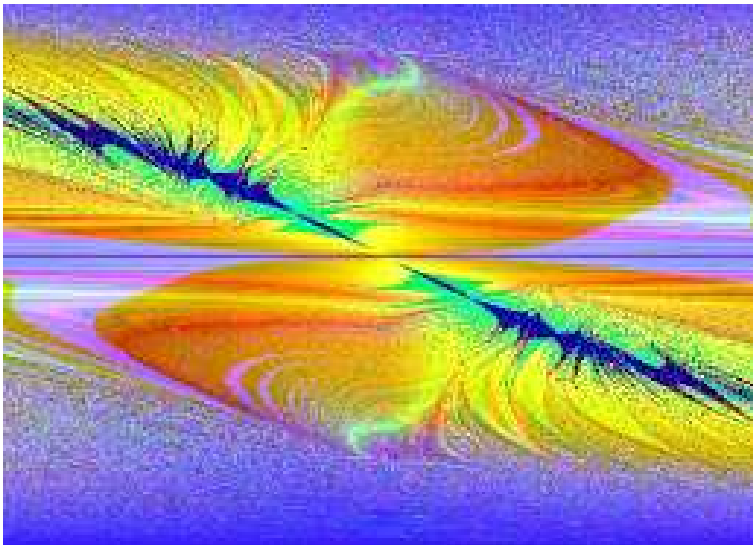
Evrimsel Sanat belki de evrimsel tasarımın yaygın olarak kullanılan en başarılı türüdür. Bu alandaki akademik araştırmalar diğer alanlardakine oranla daha az olmasına karşın, bugün Evrimsel Sanat ürünleri evrimsel tasarım sisteminin başka tiplerinin ürünlerinden daha fazladır.

Bütün Evrimsel Sanat sistemleri birbirlerine yakın olarak benzerler. Hepsi bir ilk çizgiden (rastgele ilk popülasyonlar) yeni formlar veya görüntüler üretirler. Popülasyondaki uygunluk fonksiyonunu kurmak için bir insan değerlendirme sistemine ihtiyaç duyarlar ve genellikle bu estetik görünüş ile ilgilidir. Popülasyon boyutu her neslin hızlıca değerlendirilebilmesi için oldukça küçüktür, bazen on bireyden bile az olabilmektedir. Kullanıcı arayüzleri genellikle benzer özellikler taşır, mevcut popülasyonun üyeleri ekranda bir grid formu üzerinde görülür ve böylece kullanıcı interaktif olarak bireylere istediği uygunluk değerini fare ile tıklayarak atayabilir (Şekil 4.30) (Bentley, 1999).



Şekil 4.30 : Rowbottom'un evrimsel sanat arayüz ekranı (Bentley, 1999).

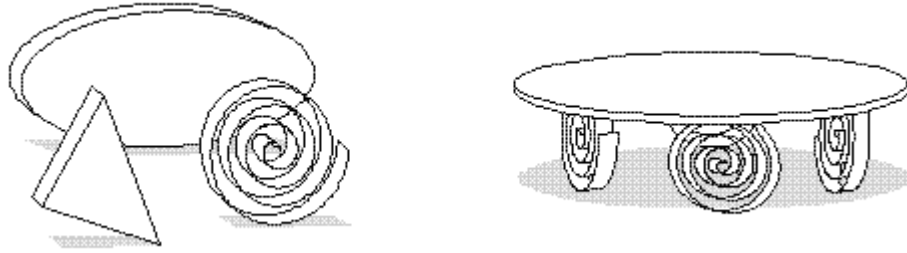
Evrimsel Sanata farklı bir örnek olarak Coimbra Üniversitesinde araştırma görevlisi olan Penousal Machado'nun NEvAr adlı evrimsel sanat yazılımının ürettiği çalışmalar verilebilir (Şekil 4.31). Sadece sin, cos gibi basit fonksiyonlar kullanılarak tasarlanan yazılım kendi kendine karmaşık sanat ürünleri geliştirebilmektedir. Çalışırken olabilecek zorlukları en aza indirgeyebilmek için NEvAR üretilen bütün popülasyonları hafızasına kaydetmekte ve istenilen karmaşıklıkta ürünler zorlanmadan üretilmektedir. Sonuçta görsel olarak ilgi çekici olan imajlar elden geçirilerek tekrar kullanılabilir (Rooke, 1999).



Şekil 4.31 : NEvAr yazılımı kullanılarak üretilmiş bir evrimsel sanat örneği (Rooke, 1999).

Evrimsel Sanat üzerine masa örneği:

Masa örneği burada farklı bir gelişme göstermektedir. Soldaki resimde masanın Evrimsel Sanat tarafından kullanılabilecek ilk şekil örnekleri görünmektedir, sağda ise bu şekillerden üretilmiş artistik masa görünmektedir (Şekil 4.32). Örnekte görüldüğü üzere sanatçının Biçim Tanım Dilini bir sabit strüktürün tasarımını geliştirmek için kullanabileceği açıktır. Bu örnekteki her artistik masa mutlaka bir elipsten ve değişik sayılarda farklı yerleştirilmiş ve döndürülmüş sarmallardan oluşmalıdır. Bu strüktür daha sonra kaç tane genin sistem aracılığıyla evrimleşeceğini ve fenotipleri üreten genlerin değerlerinin nasıl kullanılacağını belirlemektedir. Biçim Tanımlama Dili, Evrimsel Tasarım sistemleri için kalıtsal bilgileri ve embriyolojiyi tanımlar. Genlerin değerleri geliştikçe, bir sürü beklenmedik ve estetiksel olarak tatmin edici tasarımlar (bunlardan bir kısmı masa davranışları sergileyebilir) ortaya çıkabilir (Bentley, 1999).



Fenotip:

Elips, en 80, boy 50, derinlik 4, döndürme 90 derece

Spiral, yarıçap 40, kıvrım 4, derinlik 6, yatay olarak 50 derece, dikey olarak -20 derece döndürülmüş

Spiral, yarıçap 40, kıvrım 4, derinlik 6, yatay olarak 50 derece, dikey olarak -20, 90 derece döndürülmüş

Spiral, yarıçap 40, kıvrım 4, derinlik 6, yatay olarak 50 derece, dikey olarak -20, 180 derece döndürülmüş

Spiral, yarıçap 40, kıvrım 4, derinlik 6, yatay olarak 50 derece, dikey olarak -20, 270 derece döndürülmüş

Sabit Strüktür (embriyoloji) Biçim Tanım Dilini Kullanıyor:

Masa = { Elips (*en, boy, derinlik*) YZ_Döndürme (*açı*) }

{ Ayak X_Değişimi (*mesafe*) Y_ Değişimi (*mesafe*) Döndürme &Çoğaltma (*açı, #kopyalar*) }

Leg = Spiral (*yarıçap, kıvrım, derinlik*)

Genotip:

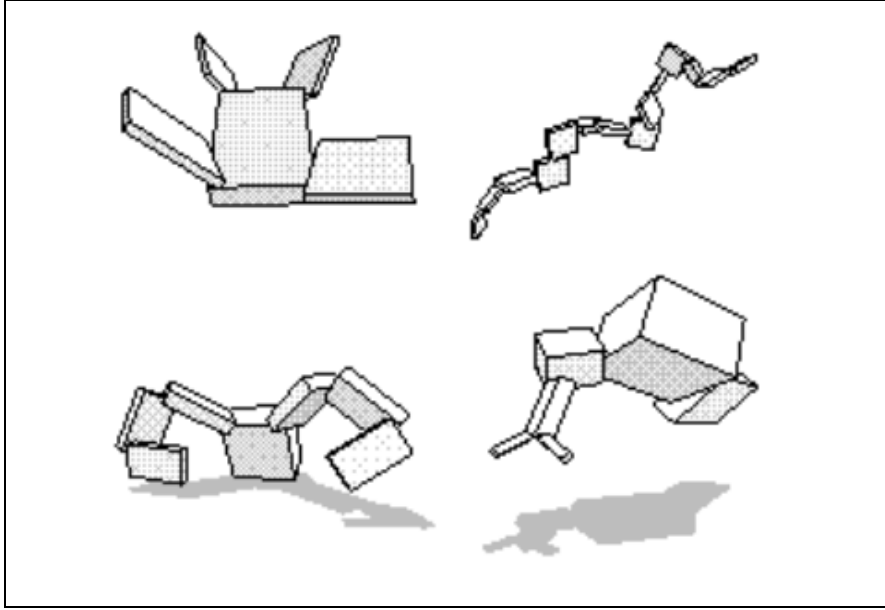
80	50	4	90	40	4	6	50	-20	90	4
<i>en</i>	<i>yükseklik</i>	<i>derinlik</i>	<i>açı</i>	<i>yarıçap</i>	<i>kıvrım</i>	<i>derinlik</i>	<i>mesafe</i>	<i>mesafe</i>	<i>açı</i>	<i>#kopyalar</i>

Şekil 4.32 : Artistik masa evrimi (Bentley, 1999).

Evrimsel Sanat farklı ve alışılmadık sanat parçaları üretmek için etkili bir yol olmasına rağmen masa gibi pratikte kullanılabilecek nesnelerin üretiminde nadiren kullanılmaktadır. Eğer elde edilen formlar işlevleri bakımından değerlendirilmeden kabul edilirse, Evrimsel Tasarım çok etkileyici örnekler sunabilmektedir.

4.2.4 Evrimsel yapay hayat formları

Evrimsel Bilgi İşleme, bilgisayar bilimindeki yeni bir görüş olan Yapay Hayat (AL) üzerinde önemli bir rol oynamaktadır. Yapay ‘zekalar’, davranış stratejileri, iletişim metotları, dağınık problem çözmek ve birçok farklı keşfedilen başlıklar genellikle Genetik Algoritmalar ve diğer evrimsel arama teknikleri kullanılırken keşfedilmiştir (Cliff, Husbands, Meyer, Wilson, 1994). Evrimsel AL formlarının içerdiği örnekler olarak: Lohn’nun tasarladığı hücresel özdevinimin kendi kendini üretmeye yetenekli olarak evrimleşmesi ve Dawkins ve Sims’in geliştirilmiş bitkiye benzer ve hayvana benzer morfolojileri verilebilir (Dawkins,1986, Sims, 1994, Lohn & Reggia, 1995).

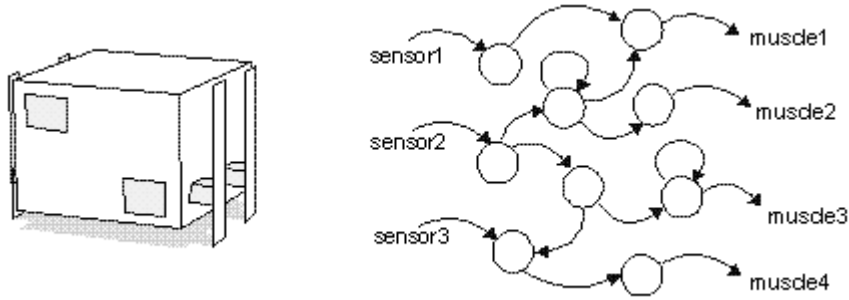


Şekil 4.33 : Sims'in evrimsel yapay hayat formu tasarımları (Bentley, 1999).

Evrimsel Yapay Hayat Formu örneği:

Evrimsel Tasarımın bu türünü açıklamak için tekrar ilk verilen 'masa' örneğini hatırlamak gerekiyor. Evrimsel Yapay Hayat Formunda statik olan bir masa yerine, şimdi sanal bir yaratık ve nesneleri etrafa ulaştırabilme kapasitesine sahip bir robot veya robot bir hizmetçi var. Şekil 4.33, bu tür tasarımların çift doğasını göstermektedir:

Fenotip:



Gövde

tip, x, y, z, en, yükseklik, blok 1 derinliği

tip, x, y, z, en, yükseklik, blok 2 derinliği

tip, x, y, z, en, yükseklik, blok 3 derinliği

.... tip, x, y, z, en, yükseklik, blok 9 derinliği

Beyin

sensör 1 nöron 1'i uyarır, en 5

nöron 2 nöron 5'i engeller, en 0

nöron 1 nöron 6'yı uyarır, en 6

sensör 3 nöron 3'ü uyarır, en 8

sensör 2 nöron 2'yi uyarır, en 4

.... nöron 3 nöron 9'u uyarır, en 4,

nöron 2 nöron 4'ü uyarır, en 3

Çıktı olarak nöron 9'un kas 4'e iletmesi

Genotip:

Kromozom 1

11	11010110	10101101	10101110	10011010	01101010	01101010	...	10001010	10001010	10001010
<i>tip1</i>	<i>xpos 1</i>	<i>ypos 1</i>	<i>zpos 1</i>	<i>en 1</i>	<i>yükseklik 1</i>	<i>derinlik 1</i>	...	<i>en 9</i>	<i>yükseklik 9</i>	<i>derinlik 9</i>

Kromozom 2

0000	10	1001	1	0110	1	...	1110
<i>nöron1 işaret</i>	<i>nöron1 tip (in/intl/out)</i>	<i>nöron1 link1</i>	<i>nöron1 link1 tip (ex/inhib)</i>	<i>nöron1 link2</i>	<i>nöron1 link1 tip (ex/inhib)</i>	...	<i>nöron9 link2</i>

Şekil 4.34 : Evrimsel yapay hayat formu (Bentley, 1999).

Evrimsel Yapay Hayat Formları genellikle form (veya formun bazı yönlerini) ve beyin evrimini içerir. Bu örnekte form veya başka deyişle 'beden' veya 'kaslar' farklı boyutlardaki blokların toplamından oluşmaktadır (bazıları 'sensör', 'beden' veya 'kaslar' olabilmektedir), 'beyin' duyu parçalarından alınan verilerin 'kas bloklarına' çıktı olarak iletilmesini sağlayan nöron ağı olarak tanımlanmaktadır. Fenotiplerin her bölümü, genotipte değişken ve uzunluk kromozomu olarak kodlanmıştır (Şekil 4.34). Uygunluk değeri, fenotiplerin en üst yüzeyi düz tutarken aynı zamanda en üst hareket kabiliyetini değerlendirmektedir. Zamanla, evrim her iki bireyin kromozomlarını sanal dünyada objeleri ve hareketleri algılayabilen hareketli bir yaratık seviyesinde geliştirmek için değişecektir (Bentley, 1999).

4.3 Genetik Algoritmaların Mimari Tasarım Örneklerinde Kullanılması

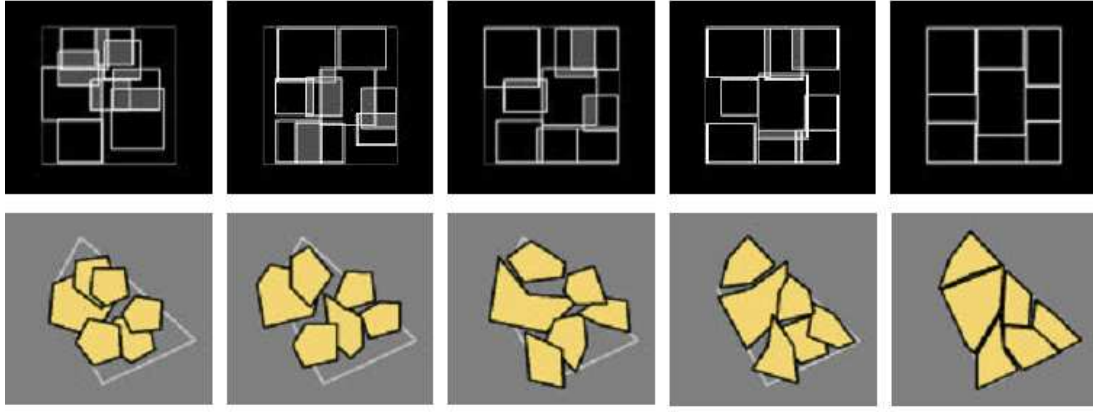
Genetik Algoritmalar birçok farklı mühendislik dalında verilen probleme çözüm olarak en iyi sonuçların popülasyonunu sunduğu için günümüzde yaygın olarak kullanılmaktadır. Genetik Algoritmalar aynı zamanda mimari tasarım örneklerinde evrimsel yaratıcı form bulma çalışmalarında, iklimsel şartlara en uygun konut tasarımı ve konut kat planı tasarımı gibi en iyileme uygulamalarında da kullanılmaya başlanmıştır. Bu bölümde Genetik Algoritmaların kullanıldığı bazı çalışmalardan örnekler sunulacaktır.

4.3.1 Konut kat planı tasarım modeli 1

Tomor Elezkurtaj'ın tasarladığı sistemde yapay evrimsel metotlar mimari kat planı tasarımında kullanılmaktadır. Tasarım girdileri birer evrimsel mekan kopyaları olarak işlem görmektedir. Bu projede evrime konu olan özellikler formla ilgili olduğu kadar aynı zamanda odaların yerleşiminin fonksiyonudur. Mimaride fonksiyonun sadece fiziksel konfor değil aynı zamanda estetik beklentileri de içerdiği anlayışı tartışmaları başladığından beri, bilgisayarın tasarımları otomatik olarak kendiliğinden evrimleştirilmesi uygun bulunmamaktadır. Bunun yerine bu modelde, kullanıcı etkileşimli olarak ekranda görülen arayüz aracılığıyla sürece dahil edilmektedir.

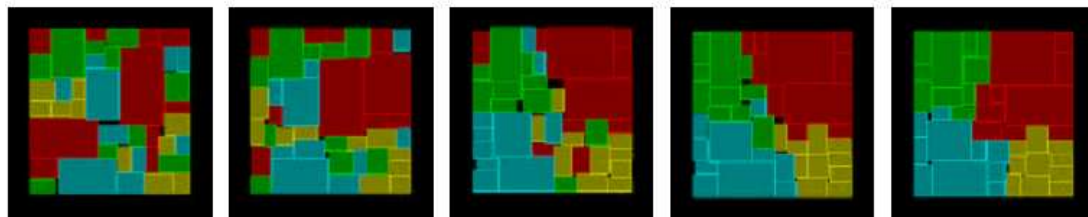
Bu kat planı tasarım modelinde sistem üç parçadan oluşmaktadır. Birinci kısımda odaları verilen boyutta ve istenilen biçimde bir bina sınırına oturtma aşaması Evrim Stratejisi (ES) ile çözülmektedir. İkinci kısımda odaların komşuluk ilişkileri gibi bir fonksiyonel gereksinime göre kontrol edilmesi, Genetik Algoritmalar tarafından sağlanmaktadır. Üçüncü kısımda ulaşılan en iyi tasarım değişkeni sonucuna fare yardımı ile müdahale olanağı sağlayan kullanıcı arayüzü vardır (Elezkurtaj & Franck, 1999).

Modelin tercih edilen şekil ve boyuttaki belirli sayıda odayı, belirli dış sınırlar içerisine uygun olacak en iyi şekilde yerleştirme görevi, istenilen oranları optimize ederken, boşlukları, çakışmaları ve dışa taşmaları en aza indirgeyerek düzenlemektir (Şekil 4.35).



Şekil 4.35 : Odaların verilen ebat ve tercih edilen biçimde, boşluklar, üst üste çakışmalar ve dışa taşmalar minimize edilerek bir sınıra oturtulması evrim stratejisi kullanılarak uygulanmıştır (Elezkurtaj & Franck, 1999).

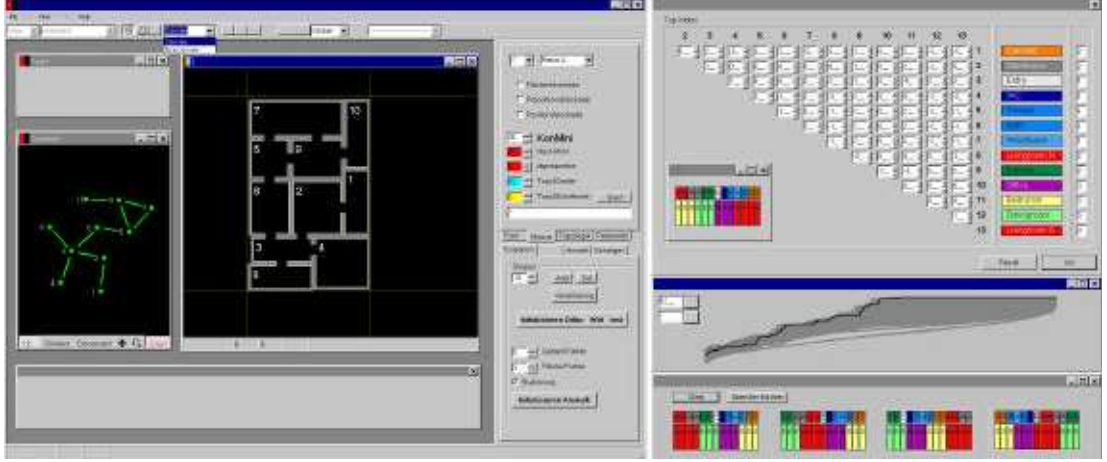
Odaları bir dış kontur içerisine oturtmak, odalar arasındaki komşuluk ilişkilerinin en iyilemesi işlemi ile karşılaştırılınca daha kolay bir görev olmaktadır. Bu en iyileme probleminin araştırılması daha karmaşıktır ve arama uzayı daha kapsamlı bir çalışma ister. Bu zahmetli görevi tamamlayabilmek için çaprazlama ve mutasyonun beraber kullanıldığı bir Genetik Algoritma kullanılmıştır (Şekil 4.36). GA tarafından uygulanan bu işlemde daha önceki odaların dizilimine bir çeşit yeniden yorumlama katılmaktadır. Komşuluk ilişkilerini en iyileme amacıyla, odalara bağlı olan fonksiyonlar değiştirilir. GA sürecinin çıktısı böylece ES'nin odaları oturtma sisteminin bir girdisine dönüşmektedir veya tam tersi de olabilmektedir (Elezkurtaj & Franck, 1999).



Şekil 4.36 : Modüllerin (odaların bölümleri) seçilen komşuluk ilişkilerine göre belirli bir dış kontör içerisinde Genetik Algoritmalar ile yeniden düzenlenmesi (Elezkurtaj & Franck, 1999).

Böyle GA ve ES gibi iki farklı çalışma sisteminden oluşan karma stratejinin benimsenmesinin altında süreci hızlandırma amacı yatmaktadır. Kullanıcı ile karşılıklı etkileşimde hız çok önemli bir rol oynamaktadır. Arayüz o ana kadar elde edilen en yüksek skorla tasarım değişkenini göstermektedir ve aynı zamanda programın çalıştığı esnada odaların konumunu, yerleşimini, biçim ve boyutlarını

değiştirme imkanı sağlamaktadır (Şekil 4.37). Bununla birlikte arayüz, odaların yerleştirileceği dış konturu ve komşuluk ilişki ağırlıklarını değiştirmeye de açıktır.

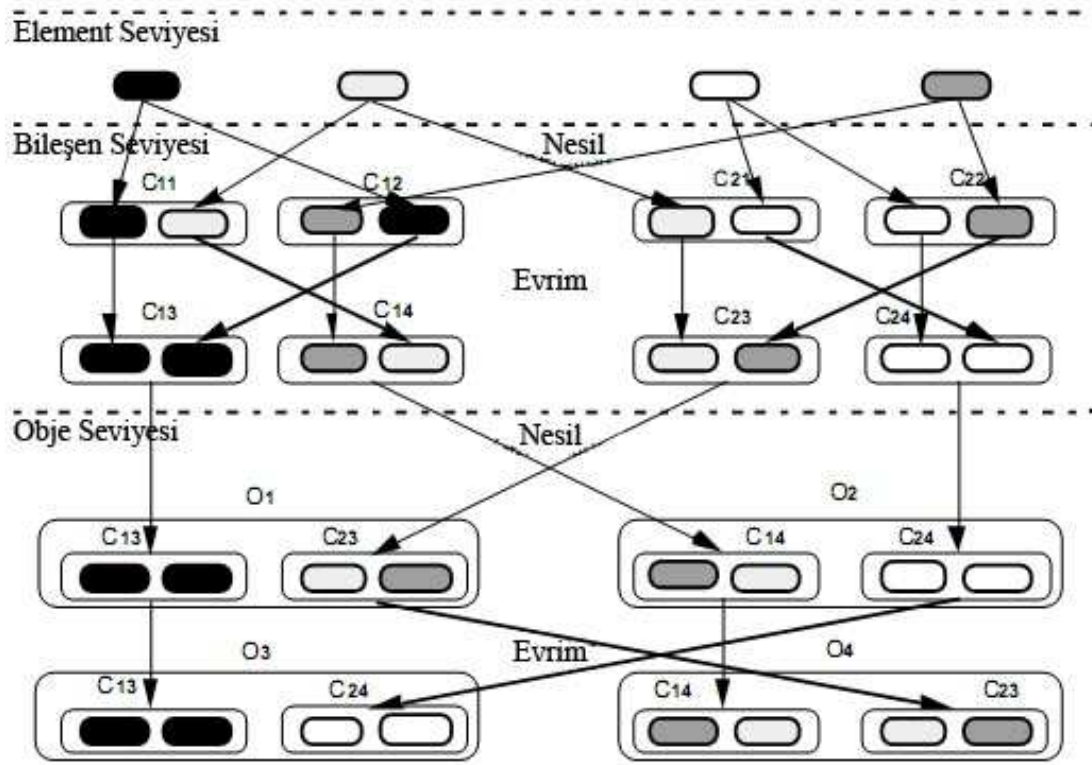


Şekil 4.37 : Kullanıcı arayüzü solda topolojiyi (ilişki bağlantılarını), ortada kat planı geometrisini, sağda ise seçilen komşuluk ilişkilerinin bağ ağırlıklarını anlatan topolojik matris gözükmemektedir (Elezkurtaj & Franck, 1999).

Bu projedeki kullanıcı ile olan karşılıklı etkileşim, makinenin ulaşamayacağı sonuçların tasarımcının yetenekleri ile elde edebilmesi açısından önemlidir. İnsan zekası veya sezgisi, karmaşıklık veya ihtiyaçların anlaşılmazlığı gibi durumlarda da başarı sağlayabilmektedir. Ayrıca, insan tasarımcı sezgisel zekanın algısı ile ne algıladığının açıklama yeteneği arasındaki boşluğu kullanır. Burada da insan zekası ile makine tarafından kazara keşfedilenler sayesinde sürpriz tasarımlara erişilmektedir.

4.3.2 Konut kat planı tasarım modeli 2

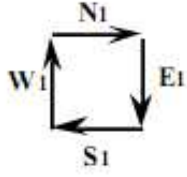
Rosenman'ın (1996) iki boyutlu, tek katlı konut kat planı üretim çalışmasıdır. Bu çalışmada uygulamalar için uygun yapı taşları olan ve tasarım grameri kurallarını temsil eden genotiplere dayanan evrimsel yaklaşımların kullanım olanakları araştırılmaktadır. Tasarım objesinin ayrışma – toplanma hiyerarşik organizasyonu, uygulamadaki paralelliklerin maksimize edilmesi ve tümleşik sorunların aşılabilmesi için kullanılmaktadır.



Şekil 4.38 : Çok seviyeli birleşme ve çoğalma (Rosenman, 1996).

Hiyerarşik organizasyon için iki yaklaşım vardır. İlki yukarıdan aşağıya, diğeri ise aşağıdan yukarıya olan yöntemdir. Burada da her seviyede ki bileşenlerin hemen üst seviyedeki bir başka bileşenler tarafından üretildiği aşağıdan yukarıya çok seviyeli yöntem kullanılmıştır (Şekil 4.38). Her seviyede önce bir ilk popülasyon üretilmektedir ve sonra popülasyondaki objeler istenilen kaliteye gelene kadar evrimleşerek yeni nesiller geliştirilmektedir. Bu popülasyonun üyeleri, gelecek seviyenin ilk popülasyonlarının bileşenlerini oluşturmak üzere seçilmektedir. Bu süreç her seviye için tekrarlanmaktadır (Rosenman, 1996).

Bu modelde konutun uzamsal hiyerarşisi ise yaşama bölgesi, eğlence bölgesi, yatak bölgesi, servis bölgesi gibi evin farklı sayıda alanlara ayrılması ile sağlanmıştır. Her bölge birden fazla odanın veya boşluğun birleşmesinden meydana gelmektedir; örneğin oturma odası, yemek odası, yatak odası, hol, banyo gibi. Farklı evler farklı odaların oluşturduğu farklı bölgelerden bir araya gelebilir. Her oda ise birkaç birim boşluktan meydana gelmektedir (Şekil 4.39). Genel olarak konut tasarımında değişmeyen tek parça birim boşluktur (Rosenman, 1996).



$$Pl(p) = (W1, N1, E1, S1)$$

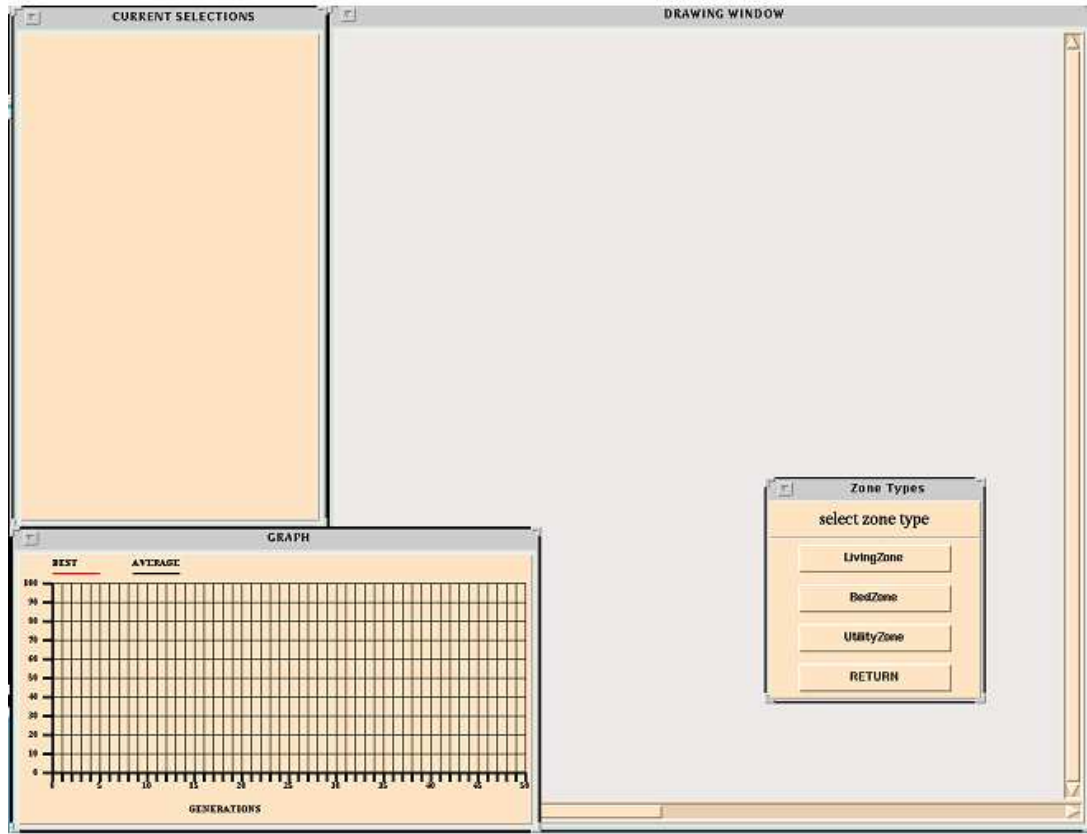
Şekil 4.39 : Odaları oluşturan birim boşluk (Rosenman, 1996).

Evrimsel tasarımda, mekanların nesilleri hiyerarşik olarak bir alt seviyedeki bileşenlerin bir araya getirilmesiyle oluşturulmaktadır. Oda seviyesinde bileşenler temel birim boşluklardır. Bölge seviyesinde bileşenler oda olmaktadır. İlk popülasyonlar her seviyede tasarım gramerine göre oluşturulmaktadır. Böyle oluşan her ilk popülasyon daha sonra evrimleşerek çözümler tasarım ihtiyaçlarına göre uyarlanmaktadır. Bu uyarlamayı oluşturabilmek için kullanılan işlemcilerden biri de çaprazlamadır. Burada evrimsel süreçte çocuk bireyler elde etmek için basit tek noktalı çaprazlama yöntemi kullanılmıştır. Seçim yöntemlerinden rulet seçimi de hangi bireylerin birleşme için seçileceğini göstermektedir (Rosenman, 1996).

Modeli oluşturan bilgisayar programı C++ and Tcl-Tk ile yazılmıştır. Kullanıcı arayüzü 3 farklı pencereden meydana gelmiştir. Sol üst pencere mevcut seçimleri, sol alt pencere popülasyondaki bireylerin seviyesini gösteren grafik penceresi, sağ pencere ise evrimsel işlemleri göstermektedir (Şekil 4.40).

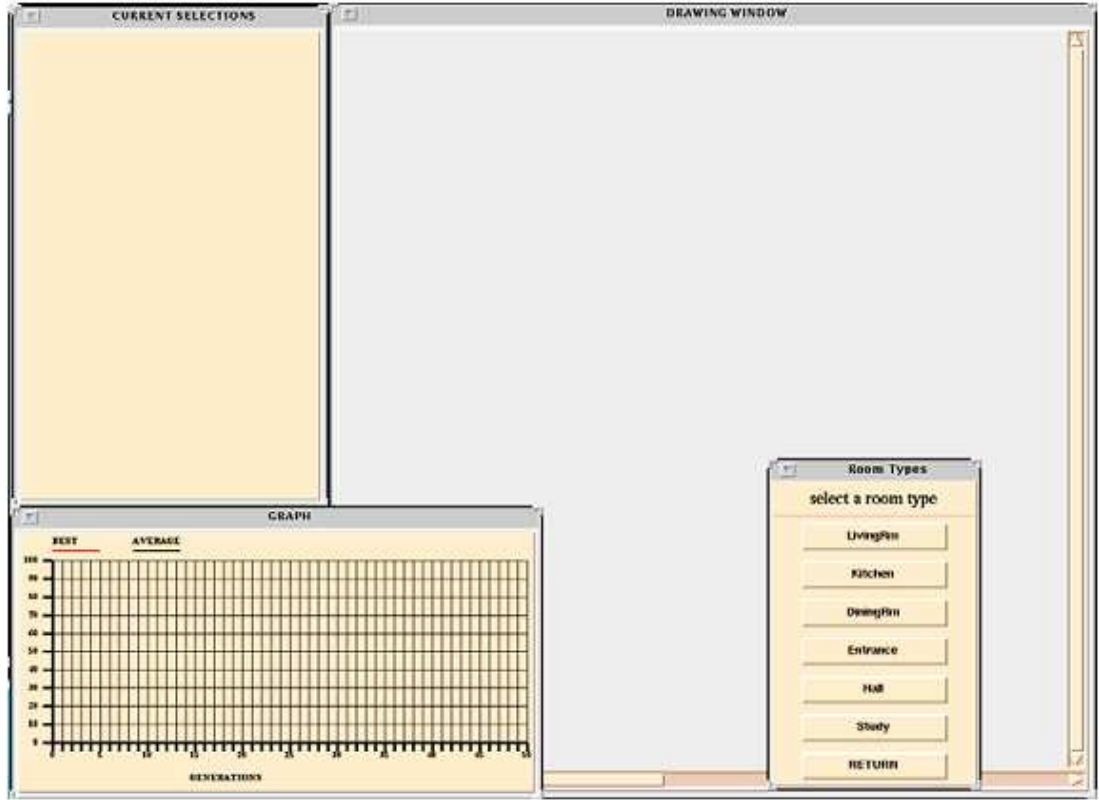
Programın Çalışmasının Örneklenmesi

Programın çalışma aşamasında arayüz panelinde, ilk önce kullanıcıdan yaşam bölgesi, yatak bölgesi ve servis bölgelerinden birini seçmesi beklenir (Şekil 4.41).

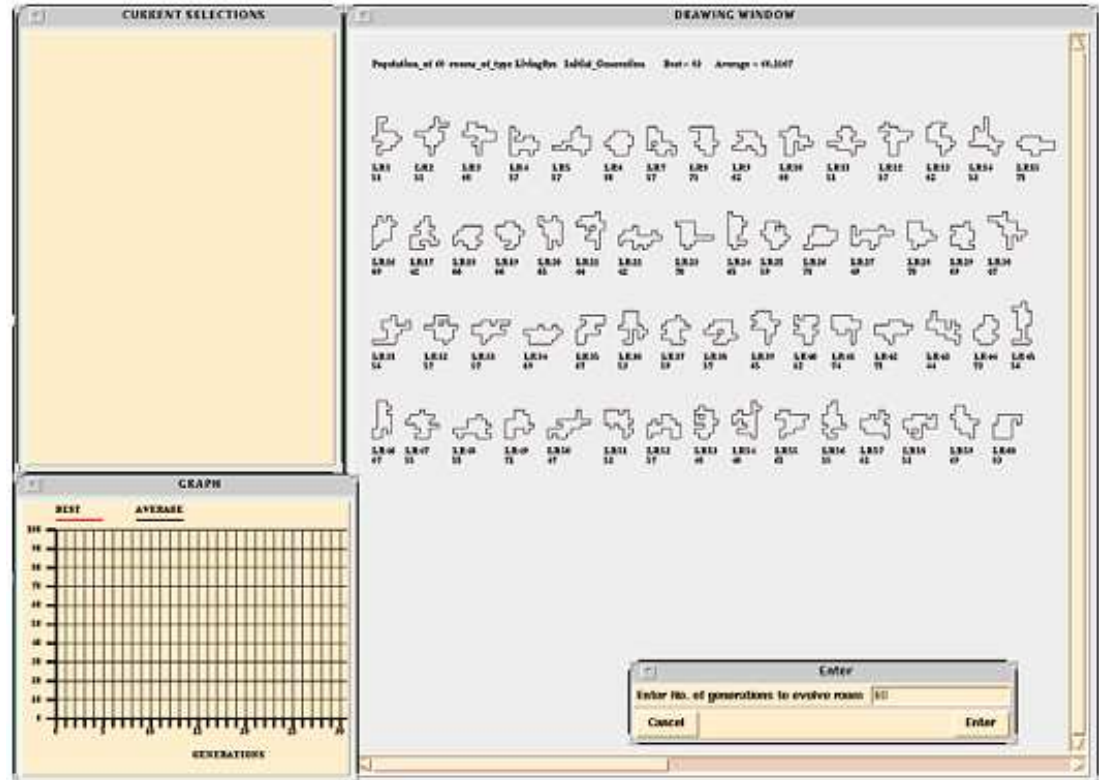


Şekil 4.40 : Kullanıcı arayüzünde bölgelerin seçimi (Rosenman, 1996).

Örnekte yaşama bölgesinin seçilmesinin ardından kullanıcıdan bölge içerisinde bulunabilecek olası odalardan birini oluşturmak üzere seçim beklenir (Şekil 4.42). Bu örnekte oturma odası, mutfak, yemek odası, giriş, hol ve çalışma odası gibi seçeneklerden oturma odası seçilmiştir. Daha sona kullanıcı arayüzünde tercih edilen oda için istenilen minimum metrekare sayısı sorulmuştur. Kullanıcı oda metrekaresini 21 olarak girmiştir. Oda metrekaresi belirlendikten sonra açılan yeni kullanıcı arayüzünde odaların oluşturulması için popülasyondaki birey sayısının kaç kabul edileceği yani ilk popülasyonda üretilecek oda sayısı tekrar kullanıcıya sorulmaktadır. Burada popülasyon sayısı 60 olarak girilmiştir.

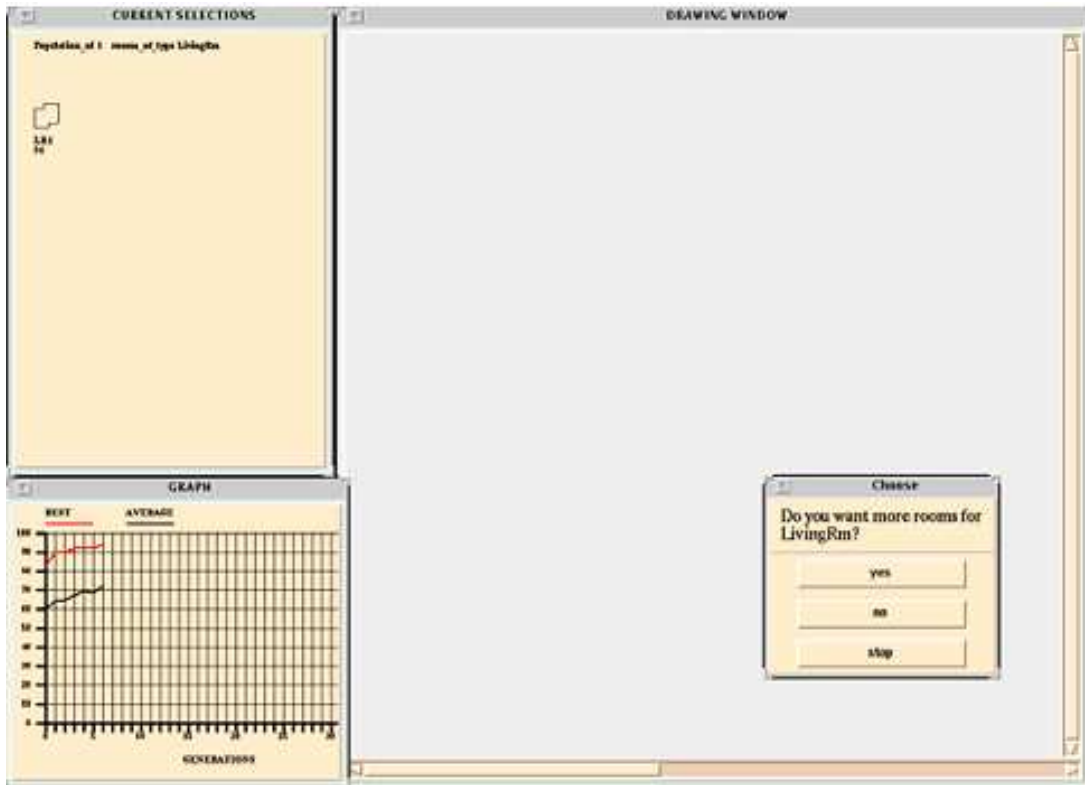


Şekil 4.41 : Kullanıcı arayüzünde yaşama bölgesinden oda seçimi (Rosenman, 1996).



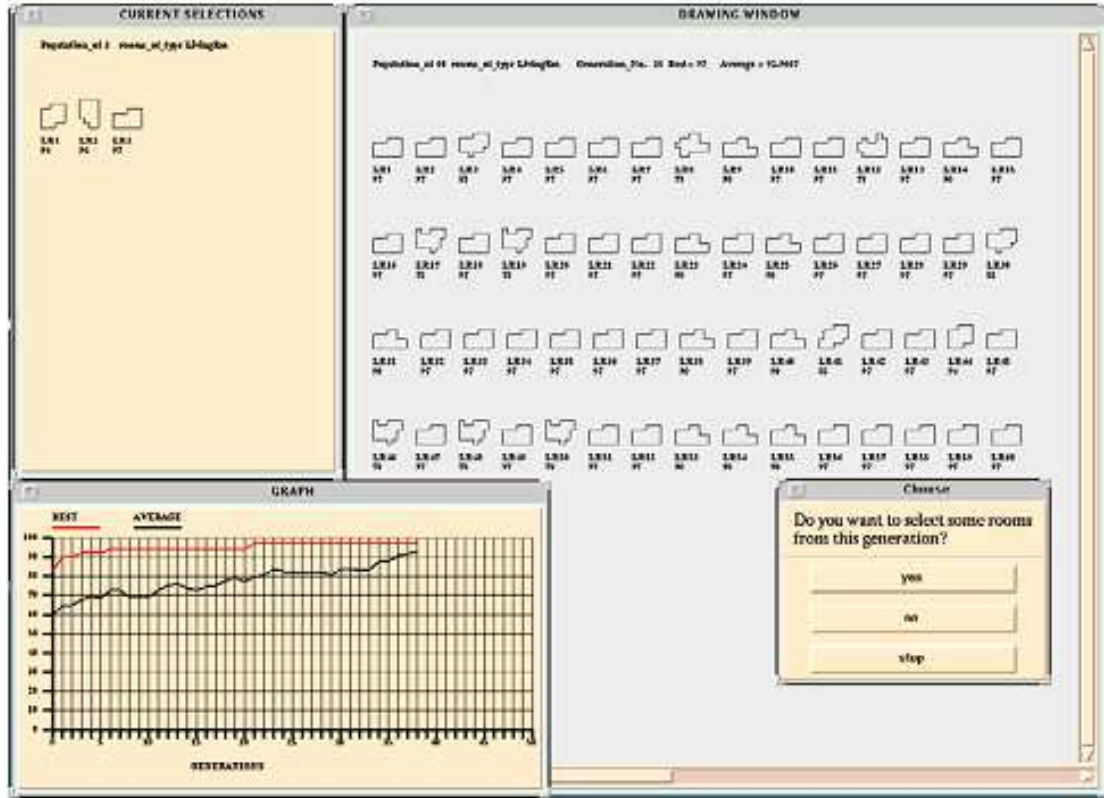
Şekil 4.42 : Oturma odası için üretilen ilk popülasyon (Rosenman, 1996).

Tercih edilen oturma odası için rastgele olarak üretilen ilk popülasyonda bütün bireylerin uygunluk fonksiyonuna göre bir değere sahip olduğu görülmektedir. Biçim grameri için uygunluk fonksiyonu, alan oranı ve biçimdeki açılarının sayısını en aza indirmektir. Odalara 0'dan 100'e kadar uygunluk fonksiyonuna göre bir değer verilmiştir. Burada uygunluk fonksiyonuna en yakın değer olarak 83 puan alan oda olmuştur ve odaların ortalaması 60.2167 olarak gözlemlenmiştir (Şekil 4.43). İlk yürütmedeki evrimleşme süreci devam ettikçe her nesilde daha farklı şekillerde odalar elde edilmiştir. 94 puan alan bir örnek de, en iyi çözüm olmaya uzak olmasına rağmen programa farklılık katması açısından seçilmiştir.



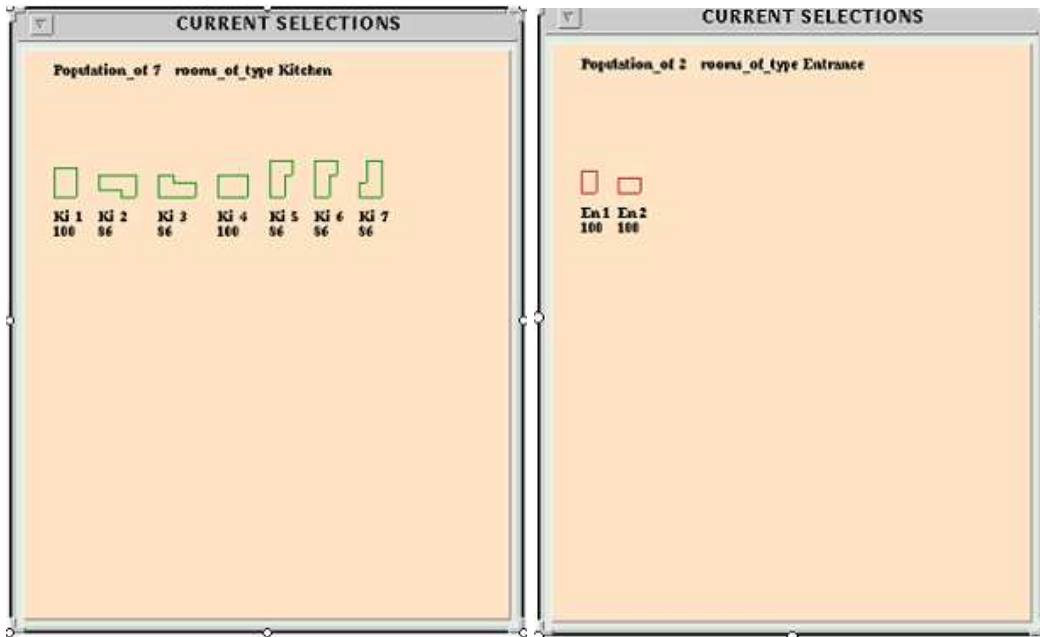
Şekil 4.43 : Nesilden bir odanın seçilmesi (Rosenman, 1996).

Şekil 4.44'te görülen nesil 38'de üretilen örnekler arasında yakınsama olduğundan dolayı ilk evrimleşme durdurulmuştur. Kullanıcı arayüzü panelinde eğer uygulamadan elde edilen sonuçları tatmin edici bulursa çalışmayı durdurabilir. İlk yürütmeden elde edilen 3 ayrı oturma odası sonucu belleğe kaydedilmiştir. Daha sonraki yürütmelerin farklı nesillerinden elde edilen toplamda birbirinden farklı 11 adet sonuç birey elde edilmiştir.



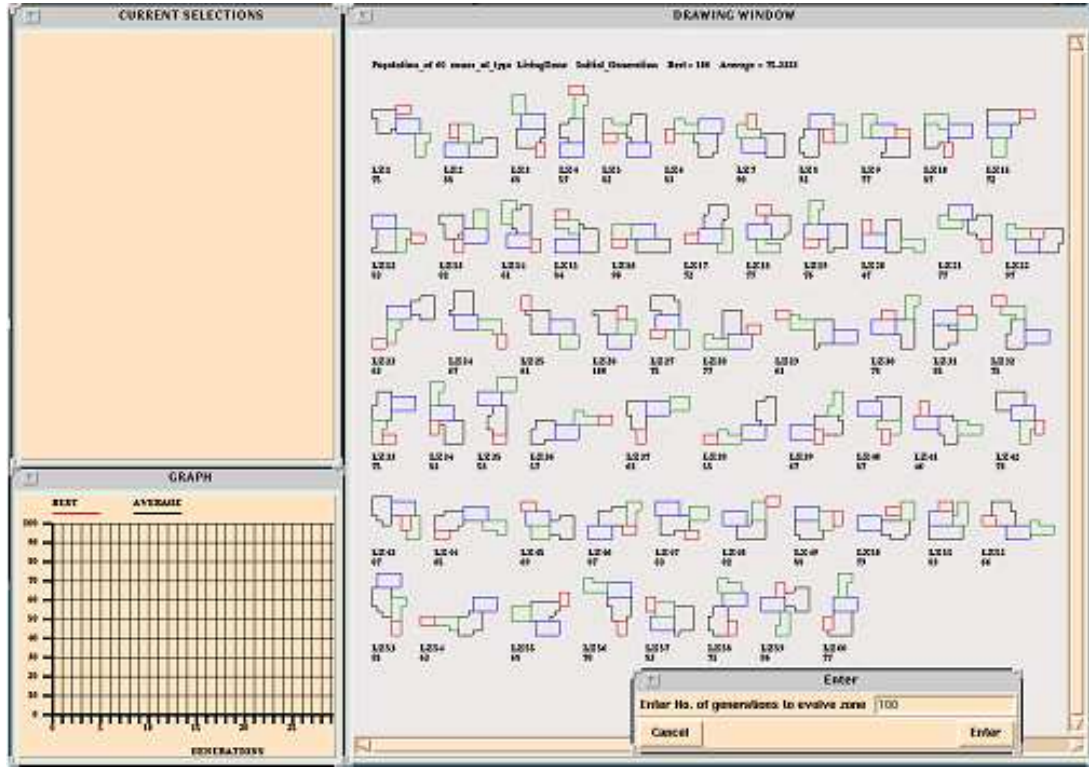
Şekil 4.44 : 38. nesilde ilk yürütmenin durdurulması sonucunda elde edilen odalar (Rosenman, 1996).

Daha sonra benzer aşamalar takip edilerek yaşama bölgesine ait 7 farklı mutfak ve 2 farklı giriş seçenekleri de elde edilmiştir (Şekil 4.45).



Şekil 4.45 : Mutfak ve giriş seçenekleri (Rosenman, 1996).

Yaşam bölümündeki bütün odalar tek tek kullanıcı beğenisi ile seçilmiş ve evrimleşmişlerdir. Sıra elde edilen odalardan yaşama bölümünü oluşturmaya gelmiştir. Bu aşamada farklı odalar seçilerek yan yana getirilmektedir. İlk nesil için 60 bireylik bir popülasyon üretilmiştir. Burada uygulanacak olan uygunluk fonksiyonu bazı odalar arası bağlantı şartlarının sağlanmış olmasıdır. Örnek olarak yemek odası ile mutfakın birbirine bağlı olması verilebilir. Bu şartları en iyi sağlayanlar ve sağlamayan bireyler (yaşama bölgesi önerisi) arasında 0'dan 100'e kadar bir değer verilmektedir ve seçimler bu değere göre yapılmaktadır. Şekil 4.46'da yaşama bölgesi için oluşturulan ilk popülasyonu göstermektedir. İlk popülasyonun ortalama değeri 75.3333'tür (Rosenman, 1996).



Şekil 4.46 : Yaşam bölgesinden üretilen ilk popülasyon (Rosenman, 1996).

Birinci yürütmenin 10. Neslinde oluşturulan yaşam bölgeleri arasında yakınsama olduğundan program durdurulmuştur. Elde edilen 4 adet yaşam bölgesi sonuçları sırasıyla 1., 2., 4. ve 8. nesilden seçilmişlerdir. Ortalama popülasyon değeri grafik penceresinden de anlaşılacağı üzere 88.5833'tür (Şekil 4.47).

Bu Genetik Algoritmaya tasarımın geometrisini direk olarak deęiřtirmeyi deęil, bir tasarımı üreten kurallar dizisini uyarlama imkanı tanımaktadır. Böylece deęiřiklięe uğramıř olup uyarlanan tasarımlar biçim grameri tarafından tanımlanan tasarım dilinde geçerli olacaktır ve tasarımcı tarafından ilk başta seçilen mimari stilistik karakterler korunacaktır.

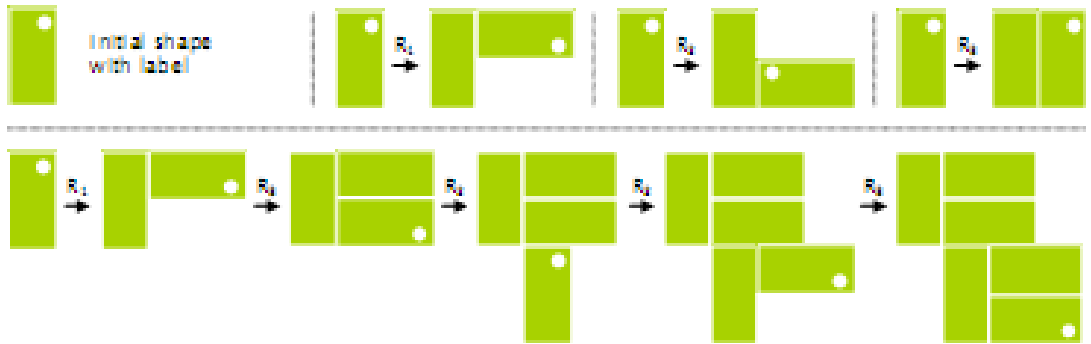
Biçim evriminin bir prototipi olan apartman bloklarının tasarımında basit biçim grameri ve bir dizi evrimsel program kullanılarak geliştirilmiştir. Denemeler farklı en iyilemeler amaçlanarak uygulanmıştır ve sonuçlar biçim grameri ve Genetik Algoritmaların kullanımının istenilen doğrultuda çalıştığı gözlemlenmiş ve tasarımcının tarzının gereklerini yerine getiren örnekler oluşmuştur. Sonuç olarak biçim evriminin bir mimari tasarım aracı olarak kullanımının uygun olduğu görülmüştür (Couchoulas, 2003).

Biçim grameri ve Genetik Algoritmaların birleşimi

Genetik Algoritmalar kullanılarak deneyimlenen evrimsel sistemler güçlü ve çok yönlü sistemlerdir. Tasarımcı için olası çözüm stratejilerinin tanımlanmasını beklemeden simültane olarak herhangi sayıda kriterin değerlendirilmesini sağlamaktadır. Alternatiflerin genişçe değerlendirilmesi tasarımların optimize edilmesine yardımcı olmaktadır. Aynı zamanda tasarımcı için her özelleşmiş tasarım problemine uygun, mantıklı tasarımlar üretmektedirler. Bir mimari projedeki fonksiyonel gereksinimler gibi ölçülebilir standartlar çözümleri bulmayı çok etkili bir şekilde sağladığı için Genetik Algoritmalar güçlü arama algoritmalarıdır. Biçim gramerleri ise formların üretiminde estetik ve strüktürel özellikler barındırmakta iyidir. Bu ikisinin karışımı olan bir sistem bir tasarım uzayında biçim gramerlerinin özelliklerinin kullanımına ve Genetik Algoritmalar bu uzayı etkili olarak kullanılır kılmaya izin vermektedir. Bu birleşim bir tasarım probleminde birçok uygun çözümler üretmektedir. Biçim grameri ve Genetik Algoritmaların birleşiminde tasarımların uygun olması için iki şart aranmaktadır. Birincisi tasarımcı tarafından belirlenen biçim gramerleri korunmalıdır. İkincisi ise, arazi sınırları ve yapı yönetmelikleri gibi mimari tasarımdaki işlevsel kısıtlamaları yerine getiren tasarımlar Genetik Algoritma tarafından seçilmelidir (Couchoulas, 2003).

Biçim kodları

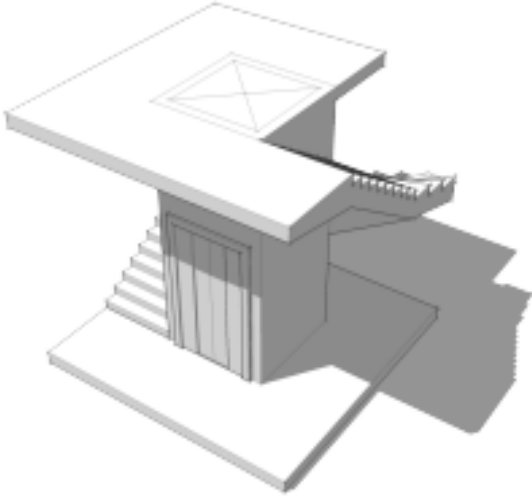
Üretim sürecinde biçim grameri kurallarının uygulanması bir ilk şekil ile başlar. Biçim grameri genellikle az sayıda biçimden ve az sayıdaki kurallardan oluşur böylece bir üretim sürecini kodlarken basit bir dizi oluşturmak yeterli olur. Bu dizi ilk hangi şekilden başlandığı ve kuralların hangi sırayla takip ettiğini gösterir (Şekil 4.48). Biçim grameri kullanılarak tasarımlar üretildiğinden beri bilinen bir dizi biçim ve bir veri yapısına herhangi bir biçim grameri kodlamak için bir kurallar seti gereklidir. Bu biçim gramerindeki veri yapısı Genetik Algoritmada genotip olarak kullanılabilir.



Şekil 4.48 : Biçim grameri kuralları ile üretilen basit bir tasarım (Couchoulas, 2003).

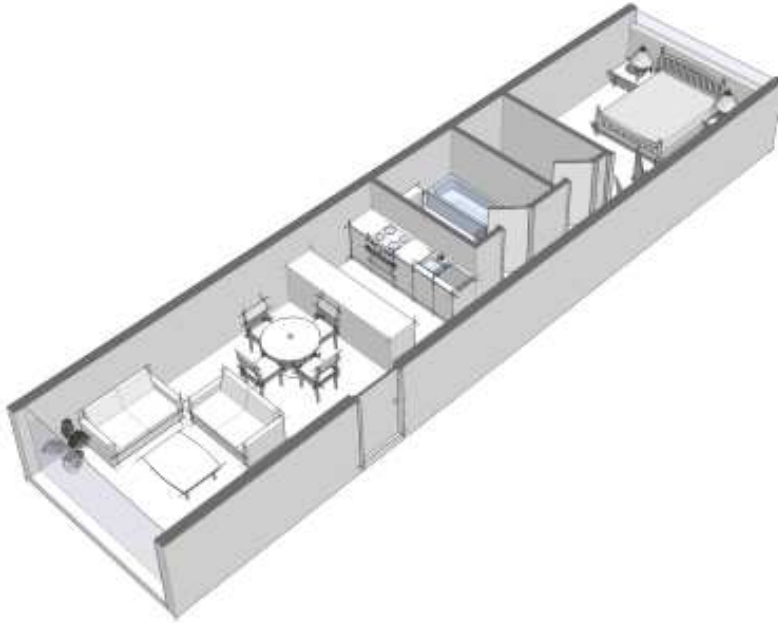
Apartman bloğu tasarımı

Tasarımlar için bir yöntem olan biçim evrimi genel olarak her tür tasarım probleminde kullanılabilir. Basit yerleşim probleminin üç boyutlu detayları biçim evrimi prototipi için örnek tasarım sorunu olarak hizmet vermeye formüle edilmiştir. Bu çalışmada çok katlı bir apartman bloğu için yenilikçi ve işlevsel tasarım konseptleri kullanılmıştır. Basitlik kavramını desteklemek için apartman blokları sadece apartman birimlerinden ve bunları birbirine bağlayan koridorlar, merdivenler, asansörler gibi yatay ve dikey sirkülasyondan oluşmaktadır. Bu tür tasarımın üretilmesi için biçim gramerinin dağarcığından sadece iki şekil yeterli olmuştur, birincisi çok amaçlı sirkülasyon birimi, diğeri ise apartman biriminin kendisidir. Sirkülasyon birimi yatayda ve düşeyde hizmet verecek şekilde yaklaşık 4m x 4m x 4m lik bir küpten tasarlanmıştır (Şekil 4.49).



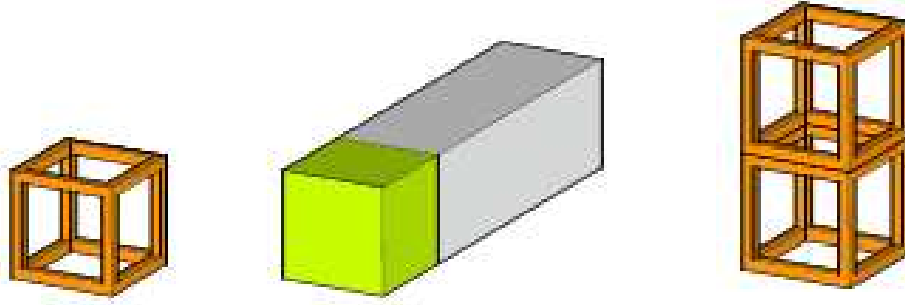
Şekil 4.49 : 4m x 4m x 4m’lik yatay ve dikeyde çalışan sirkülasyon birimi (Couchoulas, 2003).

Apartman birimi ise 4m x 4m x 16m lik geniş tek kişilik bir stüdyo daire gibi tasarlanmıştır. Bu ölçülerde olabilecek herhangi bir apartman birimi Şekil 4.50’de görünmektedir. Bu tasarımsal özellikler sonradan değiştirilebilir. Oturma odasında bulunan büyük pencerenin yeri farklı cephelere yönlendirilebilir ve apartman biriminin girişi farklı olası tasarım sonuçlarına göre birkaç metre kaydırılabilir.



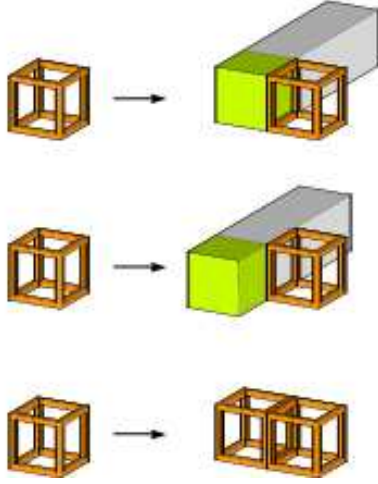
Şekil 4.50 : 16m x 4m x 4m’lik bir apartman birimi tasarımı (Couchoulas, 2003).

Bu iki detaylı tasarımın sadeleştirilmiş gösterim biçimleri sirkülasyon birimi için turuncu kafes blok ile apartman birimi için de yeşil kısmı oturma odasını temsil edecek şekilde gri bloktan oluşmaktadır. Zemin katta herhangi bir apartman birimi olamayacağı çıkarımından dolayı apartman bloğunun ilk biçimi iki sirkülasyon ünitesinin üst üste gelmesinden oluşmaktadır (Şekil 4.51).

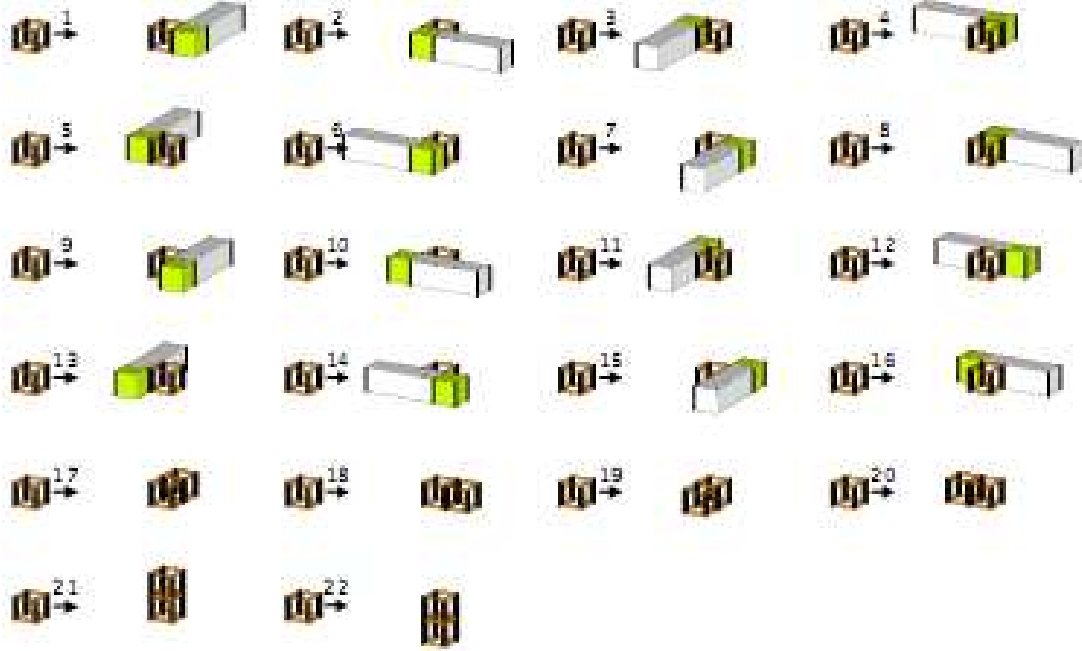


Şekil 4.51 : Apartman bloğu gramerinde birimlerin temsili ve oluşturulan ilk biçim (Couchoulas, 2003).

Apartman birimini sirkülasyon birimine bağlamanın olası farklı çözümleri, bina formunda oluşturulacak çeşitliliği arttırmaktadır. Apartman biriminin sirkülasyona oturma odasının sonunda bağlanması ilk kuralı oluşturmaktadır. Oturma odasının hemen 4 metre sonrasında bağlanması ikinci biçim grameri kuralını vermektedir. Üçüncü kural ise sirkülasyon birimlerini birbirine zincirleyerek apartman bloğunun sirkülasyon çekirdeğini oluşturmaktadır. İlk iki kural, apartman birimlerini sirkülasyona bağlamaktadır (Şekil 4.52). Çoğu kübik dönüşümlerden oluşan bu üç kural biçim grameri için uygundur ve kullanılabilir durumdadır, fakat her durum için bu geçerli değildir. Örneğin kat birimlerinin dikeyde döşenmesine olanak tanımamaktadır. Bu sebeple bu dönüşümleri de rahatlıkla sağlayabilecek toplam 22 kural belirlenmiştir (Şekil 4.53) (Couchoulas, 2003).

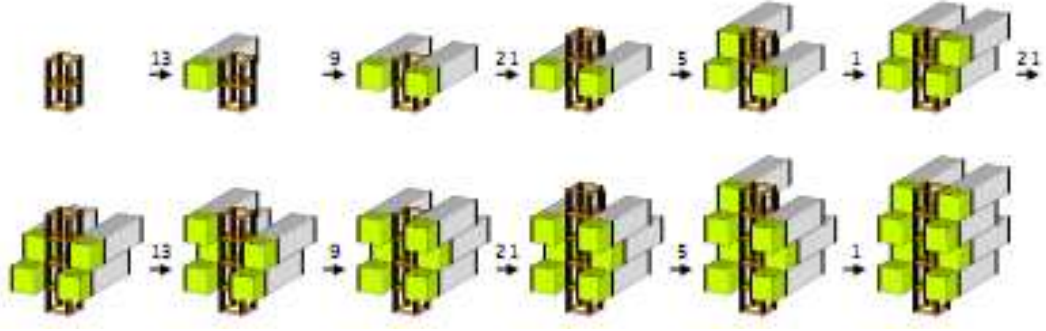


Şekil 4.52 : Apartman bloğu biçim gramerindeki üç ana kural (Couchoulas, 2003).



Şekil 4.53 : Apartman bloğu biçim gramerindeki bütün kurallar (Couchoulas, 2003).

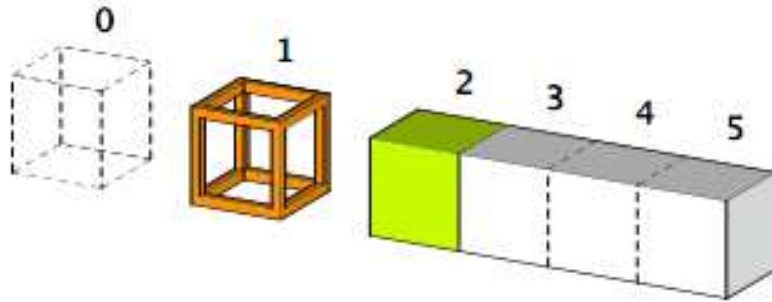
Oluşturulan 22 kural ile üretilen apartman bloğu konsept tasarımında kullanılan biçim grameri kuralları Şekil 4.54'deki örnekte gösterildiği gibi olabilmektedir.



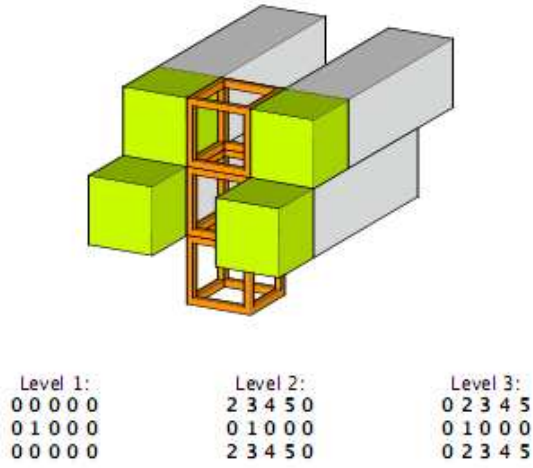
Şekil 4.54 : Bir apartmanın biçim grameri kullanılarak geliştirilen örnek tasarım dizisi (Couchoulas, 2003).

Bilgisayar uygulamasında kullanılan biçim evrimi prototipi

Biçim Evrimi C++ kullanılarak kodlanmıştır. Apartman bloğunun biçim grameri 4m x 4m x 4m kübik bir modül olarak oluşturulmuştur. Bu kolay biçim grameri ile herhangi bir çözümleyici veya üç boyutlu form tanımı için genel bir programlama sistemi olmadan da çalışabilmektedir. Boşluğu tanımlayan 0'dan başlayan değer, sirkülasyon biriminde 1 değerini almaktadır ve oturma odasına gelen kısım 2 olmak üzere diğer apartman birimi 3,4 ve 5 olarak anılmaktadır (Şekil 4.55). Böylece oluşturulan her dizi kendi içerisinde üç boyutlu olarak tanımlanabilmektedir (Şekil 4.56).



Şekil 4.55 : Apartmanın bloğu biçim gramerinde her kübik modülün numaralandırılması (Couchoulas, 2003).



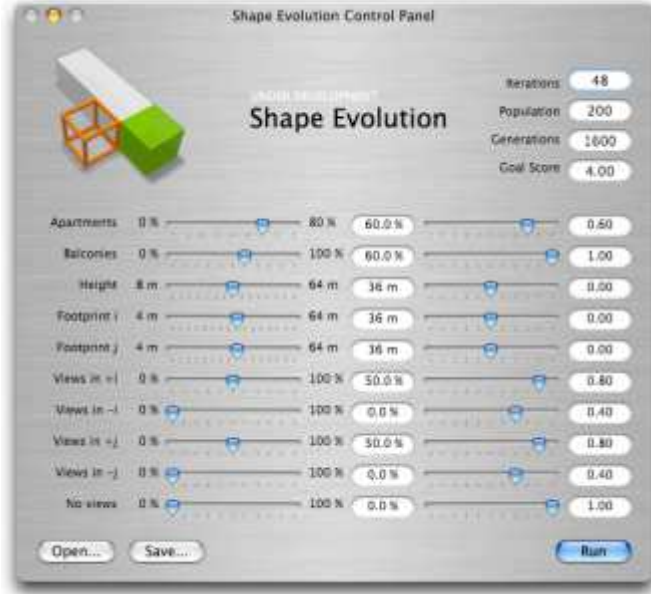
Şekil 4.56 : Apartmanın bloğu biçim gramerinde bir örneğin değerlerinin dizi olarak gösterilmesi (Couchoulas, 2003).

Yapılan çalışmaların kullanıcıya daha iyi sunulabilmesi için biçim grameri üç boyutlu diziyi sanal gerçeklik diliyle VRML dosyasına dönüştürmektedir. Böylece kullanıcı yapılan çalışmayı geometrik olarak bilgisayar ekranında farklı açı pencereleriyle görebilmektedir (Şekil 4.57). Bu arayüz yazılımı aracılığıyla kullanıcı, modeli istediği gibi çevirebilmekte ve eş zamanda sanal gerçeklikte gezinti yapabilmektedir.



Şekil 4.57 : VRML ile oluşturulan kullanıcı arayüzü (Couchoulas, 2003).

Bir sonraki aşamada ise amaçlanan kullanıcı destekli optimizasyona göre tasarımdaki ilk popülasyonun evrimleşmesi uygulaması başlar. Bu kullanıcı odaklı arayüzde evrimleşmenin karşılaştırılabilmesi için, kullanıcıdan apartman blok sayısı, alan, yükseklik, hacim, balkonlar, gibi belirli girdilere bir en iyileme değeri ataması beklenir (Şekil 4.58). Elde edilen tasarımların bu değerlere yaklaşması koşul olarak belirlenecektir.

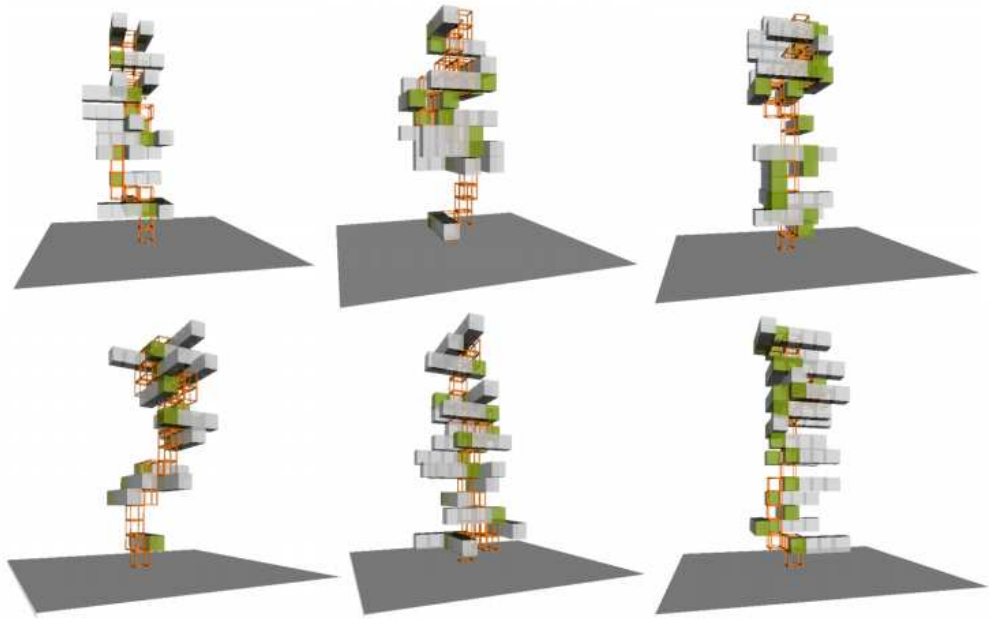


Şekil 4.58 : İsteklerin belirlendiği en iyileme arayüzü (Couchoulas, 2003).

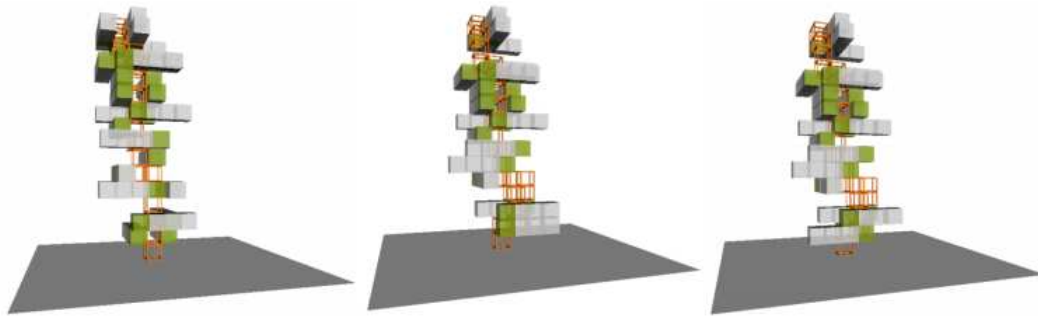
Biçim evrimi uygulamaları basit ikili sistemin turnuva seçim yöntemi ile seçilerek iyi olan bireyler gelecek nesillere aktarılmaktadır. Burada da popülasyondan rastgele seçilen iki bireyden, değerleri karşılaştırılınca daha yüksek skora sahip olan birey ilk popülasyonla birey sayısı aynı olan ara bir popülasyona alınmaktadır. Bu işlem ara popülasyon dolana kadar devam ettirilmektedir. Ara popülasyondaki yüksek skorlu tasarım kopyaları ve az ya da hiç düşük değerli tasarımın olmaması, en uygun bireylere yüksek çoğalma kapasitesi vermektedir.

Çaprazlama ise biçim evrimindeki genotip basit bir dizi ve biçim kodu olduğu için basit tek noktalı olarak uygulanmaktadır. İki ebeveyn genotip rastgele bir noktadan kırılarak, kırılmanın olduğu kısımdan yer değiştirmektedirler, böylece yeni bireyler oluşmaktadır. Mutasyon gerekli görüldüğü taktirde kullanıcı tarafından belirlenen mutasyon oranıyla popülasyondaki her genotipin her bitine uygulanmaktadır.

Biçim evrimi kullanılarak elde edilen tasarımlarda, kullanıcı tarafından yüksekliğin maksimum skorda olması istendiği takdirde elde edilen sonuçlar Şekil 4.59'da görülmektedir. Aynı yürütmede elde edilen mutasyon oranı 0.05 olan, 459., 462. ve 461. nesillerden en yüksek değere sahip üç kule arasındaki tasarımsal benzerlikler göze çarpmaktadır (Şekil 4.60). Bu da yürütmenin belirli bir aşamasından sonra Genetik Algoritmalarla gözlemlendiği üzere, tasarımlar arası yakınsama oluştuğunu göstermektedir (Couchoulas, 2003).



Şekil 4.59 : Uzunluk maksimum değerde olunca elde edilen çeşitli kule tasarımları (Couchoulas, 2003).



Şekil 4.60 : Aynı yürütmenin 459, 462 ve 461. nesillerinin en yüksek değerli tasarım örneklerinin birbirine yakınsaması (Couchoulas, 2003).

Bilgisayar destekli üretilen biçim grameri tasarımları daha sonra malzeme, strüktür gibi konularda geliştirilerek gerçek mimari tasarımlarda kullanılabilir (Şekil 4.61).



Şekil 4.61 : Biçim evrimi ile elde edilen geliştirilmiş konut bloğu tasarımı (Couchoulas, 2003).

Evrimsel yaklaşımlara dayalı Genetik Algoritmaların bilgisayar teknolojilerinde kullanımıyla beraber bir probleme ilişkin olası çözümün bir veya birkaç tane değil bir popülasyondaki birey sayısı kadar olabileceği ortaya konulmuştur. Mimari tasarımda da doğru çözüm sadece bir taneden ibaret değil, sonsuz tasarım olasılıklıdır. Doğadan esinli evrimsel yaklaşımlarla elde edilen ve bu bölüm içerisinde tanıtılan mimari tasarımlardan da anlaşılacağı üzere Genetik Algoritmalar, en iyileme mantığı ile geliştirilen olasıklık sonuç popülasyonları ile farklı tasarım problemlerine cevap verebilmektedir.

5. GENETİK ALGORİTMAYA DAYALI KİTLESEL BİREYSELLEŞTİRME AMAÇLI KONUT TASARIM MODELİ

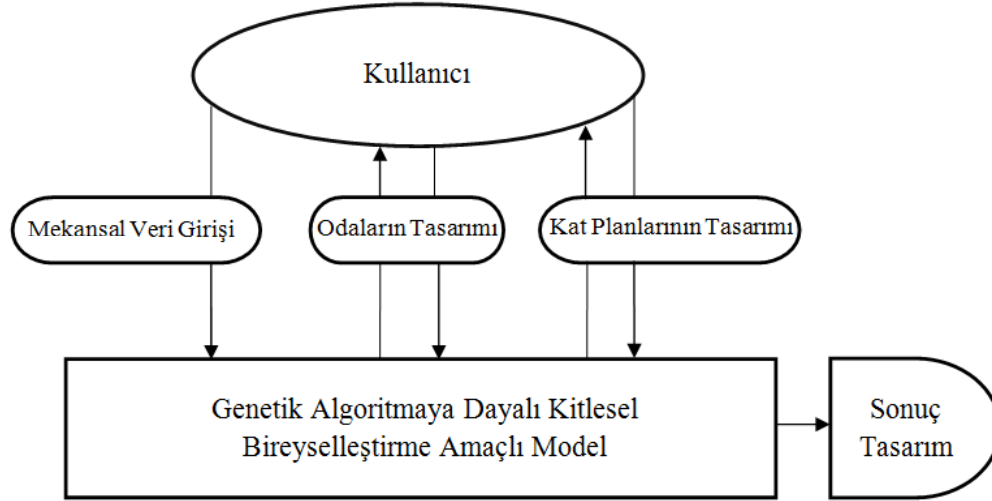
Mimari tasarımın bilgisayarlar aracılığı ile iki boyutlu ve üç boyutlu sunum çalışmalarında kullanılmaya başlanmasından sonra yazılım ve donanım teknolojilerinde önemli gelişmeler olmuştur. Gerçek anlamda ‘yaratıcı’ mimari tasarımda bilgisayar teknolojilerinin kullanılması ise 4. bölümde de açıklandığı üzere son yirmi beş yıldır tartışılan bir konudur. Bu çalışmalarda amaçlanan bilgisayarı mimari tasarımda aktif bir şekilde kullanarak yaratıcı, etkili ve hızlı sonuçlar elde etmektir.

Tez kapsamında, kullanıcı isteklerinin ön planda tutulduğu kitlesel bireyselleştirme odaklı, kullanıcıdan gelen veriler ile istenilen en iyi çözümü bulmak için Genetik Algoritmalar kullanılarak konut kat planı tasarlayan bir model oluşturulmuştur. Geleneksel yöntemler ile tasarımcının tüm tasarım uzayını, tarayarak seçenek çözümleri üretmesinin ve bunlar arasından uygun çözümleri bulmasının çok olanaklı olmadığı açıktır. Buna karşın Genetik Algoritmalar ile çalışan modelde kullanıcı tercihleri girilip, kısıtlayıcı isteklerin her biri aynı seviyede tutularak en iyileme yöntemi gerçekleştirilmektedir.

5.1 Geliştirilen Modelin Amacı ve Tanıtımı

Çalışmada, uzman tasarımcı olmayan kullanıcının konut ihtiyaçlarının belirlenip, sistematik bir şekilde konut kat planı alternatiflerini üreten bir tasarım aracı geliştirilmiştir. Bu sebeple modelin başlangıç ve ilerleyen diğer arayüzlerinde, kullanıcıdan boyutsal ve mekansal bilgi girişi istenilmektedir. Şekil 5.1’de görüldüğü gibi kullanıcının arazi bilgileri ve oluşturulacak odaların ebatları, yönleri ve komşuluk ilişkileri gibi topolojik bilgileri alan model, bu veriler doğrultusunda tek tek oda biçimlerini Genetik Algoritma kullanarak oluşturmaktadır. Genetik Algoritma işlemcileri aracılığıyla yaratılan oda popülasyonları, kullanıcının tercih ettiği üretim sayısına bağlı kalınarak üretilmekte ve kullanıcı ile etkileşimli bir şekilde seçilmesi için arayüzde sunulmaktadır. Bütün odaların kullanıcının

beğenisine sunularak seçiminden sonra farklı her oda için birçok sonuç tasarım elde edilir. Daha sonra bu odalar kat planı oluşturmak amacıyla, bir araya getirilir ve programın arayüzünde verilen kısıtlamalar ile elenerek kat planı şemalarına dönüştürülür. Oluşan kat planları iki ve üç boyutlu olarak kullanıcıya sunulur ve kullanıcı bunlardan birini seçerek program sürecini sonlandırır.



Şekil 5.1 : Genetik Algoritmaya dayalı kitlesel bireyselleştirme amaçlı modelin kullanıcı ile etkileşimli tasarımı.

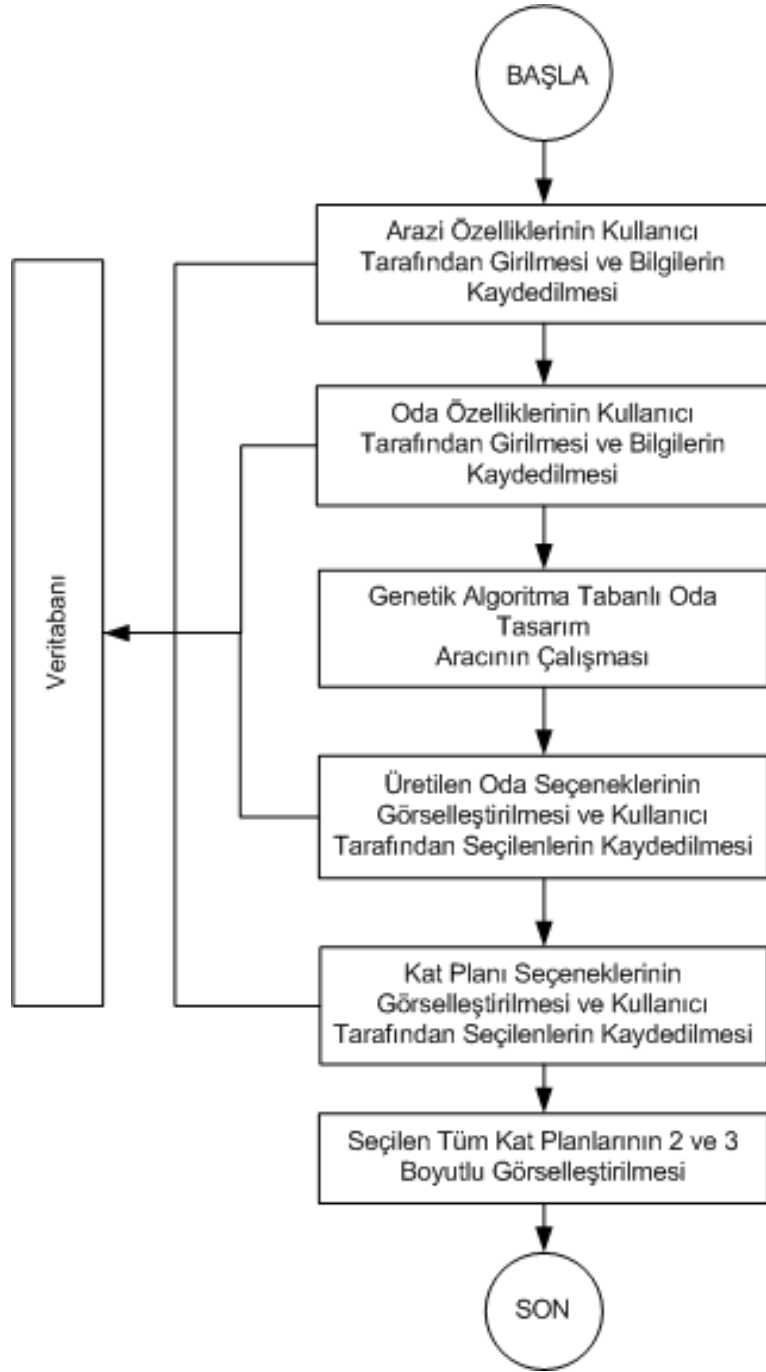
Eğer kullanıcı 3 katlı bir konut tasarlamayı seçtiyse, her kat için bu üretim çalışmaları yürütülüp oluşturulan forma, arayüzde döndürme işlemi uygulanarak farklı açıdan bakılabilir. Aynı zamanda çok katlı bir apartman bloğunun herhangi bir katı tasarlanmak istenirse model aracılığı ile belirlenen sirkulasyon boşluğu her katta sabit kalmak koşulu ile bir tasarım yapılabilir. Her katta tek dairenin olduğu düşünülerek geliştirilen modelde, diğer katlar da kullanıcıların istekleri doğrultusunda oluşturulduktan sonra kütlenin bütünü formu gözlemlenebilir. Model aracılığı ile oluşturulan birden çok katlı binanın bütün kat planlarında sirkulasyon boşluğu düşey akışı sağlamak amacıyla sabit bırakılmaktadır. Bu sabit kısım kitlesel bireyselleştirmenin üretim türlerinden olan modüler hale getirmenin bileşen paylaşırma modülasyonu (component-sharing modularity) özelliklerini de taşımaktadır. Bu ortak kullanılan sabit modül çevresine diğer odalar farklı biçimlerde üretilip birleştirilerek, kütle tasarımına hareket kazandırılmaktadır.

5.2 Modelin Algoritması

Genetik Algoritmaya dayalı kitlesel bireyselleştirmeye amaçlı model, MATLAB yazılım geliştirme aracı kullanılarak ortaya konmuştur. MATLAB, çoğunlukla mühendislik, finans gibi matematiksel araçlara yoğun biçimde başvurulmuş alanlarda kullanılan, matematiksel işlemleri, analizleri ve görselleştirmeyi kolay ve etkili bir biçimde gerçekleştiren oldukça kapsamlı bir araçtır. Diğer yazılım dillerine göre üstünlüğü, bir yazılımcıya yönelik araçlar sunmak yerine, yazılım uzmanları dışında bilim ile uğraşan insanlara hızlı ve etkili veri toplama, düzenleme, analiz etme ve sonuçları görselleştirme imkanı sunmasıdır. Ayrıca MATLAB dahilinde oldukça büyük bir araç veritabanı bulunmaktadır; hemen her türlü fonksiyon daha önceden tanımlanmış ve kolay kullanımı için ayrıntılı bir biçimde dosyalanmıştır. Bu nedenle bilimsel çalışmalarda sıklıkla kullanılan MATLAB yazılım aracı bu çalışmada da etkin bir biçimde kullanılmıştır.

Model tasarlanırken genel olarak tümevarım yaklaşımı izlenmiştir. Hiyerarşi genlerin oluşturduğu en küçük birim olan fenotiplerin uygunluk fonksiyonuna göre bir odayı oluşturmak için bir araya gelmesinden sonra odaların bir kat planı oluşturmasına kadar gitmektedir. Böyle bir tasarım için fenotiplerin oluşturduğu birim kareye değişmeyen tek sabit birim denilebilir. Bu hiyerarşiye göre odaların biçim grameri, sabit birimin Genetik Algoritmaya göre şekillenmesi ile ortaya çıkmaktadır.

Geliştirilen programın genel bir akış şeması Şekil 5.2’de sunulmuştur. Program, kullanıcıya veri girişi olanağı sağlayan ve sonuçları görselleştiren arayüzler, oda şekillerinin ve kat planlarının üretilmesini sağlayan algoritmalar ile sonuçları daha sonra kullanmak üzere saklayan veritabanı bileşenlerinden oluşmaktadır. İlerleyen bölümlerde öncelikle Genetik Algoritmaya dayalı oda şekillerinin tanımlanması ve üretilmesi işlemlerini gerçekleştiren fonksiyonlar hakkında bilgi verilmiş, ardından oluşturulan oda şekilleri ve kullanıcıdan alınan girdiler kullanılarak kat planlarının oluşumunu sağlayan fonksiyonlar tanıtılmıştır. Daha sonra ise, kullanıcı veri girişi ve sonuçların sunulduğu arayüzler örnek uygulama üzerinden tanıtılmıştır.



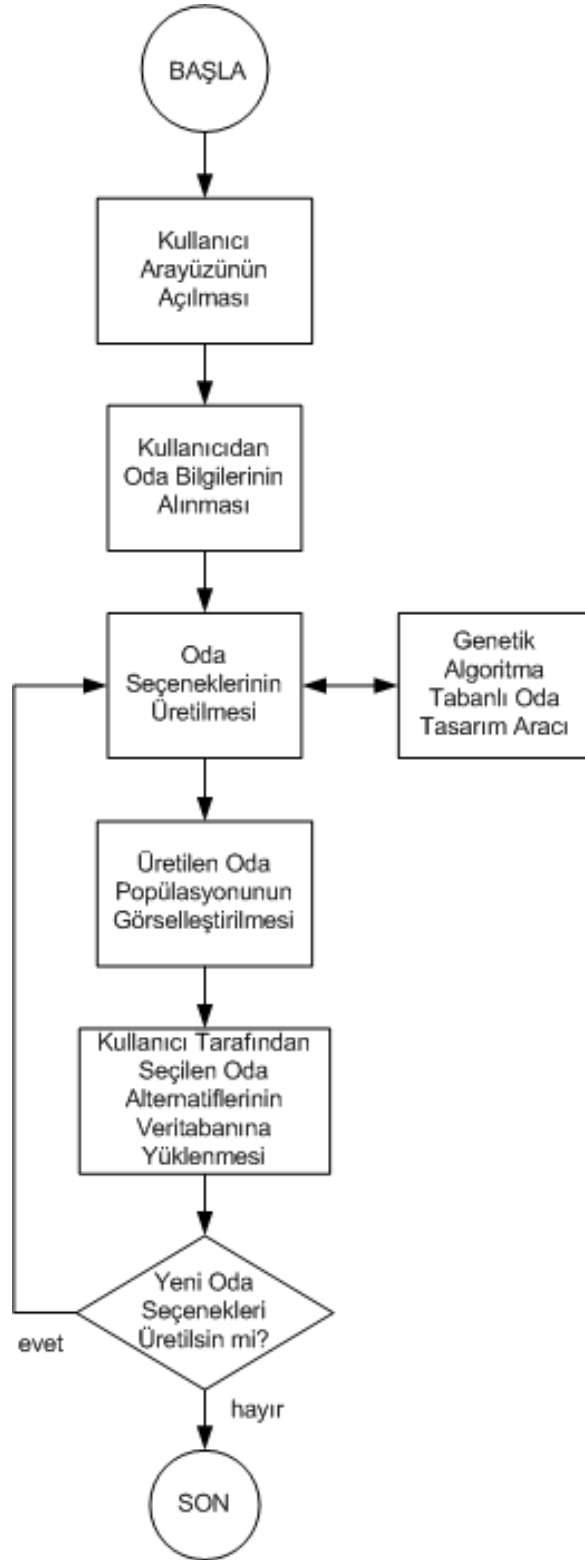
Şekil 5.2 : Genetik Algoritmaya dayalı kitlesel bireyselleştirme amaçlı modelin akış diyagramı.

5.2.1 Genetik Algoritma ile oda biçimlerinin belirlenmesi

Oda biçimlerinin belirlenmesi işleminde Genetik Algoritmanın rutinleri uygulanmıştır. Bu bağlamda oda biçimlerinin oluşturulması işlemini gerçekleştiren fonksiyonlara ilişkin akış diyagramı Şekil 5.3'te verilmiştir. Programın çalıştırılmasıyla Genetik Algoritmaya özgü parametreler belirlenir. Bunlar, Genetik Algoritmanın işleyişi sırasında kullanılan popülasyon boyutu, elitizm faktörü,

çaprazlama ve mutasyon oranı gibi parametreler olup deneyler sonucu en uygun değeri belirlenerek kaydedilmiştir. Ardından, ileri bölümlerde tanıtılacak olan kullanıcı arayüzü vasıtasıyla kullanıcıdan odaların alanları, yönleri ve komşuluk ilişkileri ile ilgili parametreler alınarak Genetik Algoritma tabanlı oda tasarım aracına iletilir. Aracın çalışması sonucu üretilen sonuçlar kullanıcıya sunulur ve kullanıcı müdahalesi ile veritabanına kaydedilirler. Bu döngü belirlenen kriterler sağlanana kadar veya kullanıcı elde edilen oda biçimlerinden memnun olup durdurana kadar devam eder. Oda biçimlerinin belirlenmesi ile ilgili tüm algoritmalar Genetik Algoritma tabanlı oda tasarım aracında toplanmıştır.

Genetik Algoritma kullanılarak konuta ait oda bileşenlerinin üretilmesi söz konusu olduğunda, algoritmanın en temel bileşenleri olan gen ve kromozomların belirlenmesi gerekir. Seçilen gen yapısı çalışılan tüm uzayda tanımlanabilir ve tek olmalıdır. Yani bir odanın en küçük bileşenine gen dersek, genlerin tüm varyasyonları yine bir gen yani odaya ilişkin bir bileşen tanımlamalı ve bir genin taşıdığı bilgiyi farklı bir gen taşımamalıdır. Ayrıca Genetik Algoritma yaklaşımında kromozomlar arası gen transferi, genlerin mutasyona uğraması kavramlarının sağlıklı bir biçimde işleyebilmesi için gen yapısının değişmemesi, tüm kromozomlarda korunması gerekir. Benzer şekilde, genlerin birleşiminden oluşan kromozomlar da çalışılan tüm uzayda tanımlanabilir ve tek olmalıdır. En küçük birimlerin birleşimi ile oluşan oda yapısı bir kromozom ile temsil edilecek olursa, kromozomun taşıdığı bilginin değişmesi farklı bir oda biçimine neden olacak, ayrıca bir odayı temsil eden kromozom tek olacak, yani aynı odanın farklı bir genotipi tanımlanamayacaktır.



Şekil 5.3 : Modelde oda biçimlerinin oluşturulmasına ilişkin akış diyagramı.

Her bir gen bir birim karelik bileşeni temsil etmek üzere, i 'nci gen, birimin 2 boyutlu düzlemdeki konumunu belirten X ve Y koordinatları cinsinden şu şekilde tanımlanabilir:

$$\text{Gen}(i) = \{\text{Birim}(i).\text{Koordinat_X}, \text{Birim}(i).\text{Koordinat_Y}\}, i=1,2,\dots,N$$

Kromozom odaya ait genotipi temsil ettiğinden, N birim karelik bir odaya ait kromozom, N adet genin birleşiminden oluşmaktadır. Buna göre N birim kareden oluşan j 'nci odaya ait kromozom yapısı şu şekilde olacaktır:

$$\text{Kromozom}(j) = \{\text{Gen}(1); \text{Gen}(2); \dots; \text{Gen}(k)\}$$

$$= \{\text{Birim}(i).\text{Koordinat_X}, \text{Birim}(i).\text{Koordinat_Y}\}, i=1,2,\dots,N$$

Bir kromozomun bir oda biçimini temsil edebilmesi için boyut koşulunu sağlamasının yanı sıra birimlerin birbiri ile kesişmemesi ve birimler arası kesinti oluşmaması gerekir. Bu yüzden N birim karelik odayı temsil eden bir kromozomun oluşturulması N adet birbirinden bağımsız genin birleşimi şeklinde düşünülemez. Kromozoma eklenen her bir genin daha öncekiler ile yukarıda belirtilen uyumu sağlaması gerekir. İlk popülasyonu oluşturacak N uzunluklu kromozomun oluşumunu sağlayan kod parçası aşağıda sunulmuştur. Fonksiyonun girdisi birim kare cinsinden oda alanı, çıktısı ise N birim karelik bir odayı oluşturan her birimin koordinatları ile odanın dış duvarlarının koordinatlarıdır. Dış duvarlar, kendisini oluşturan tüm birim uzunluklu doğru parçalarının iki uçlarındaki noktaların koordinatları ile aşağıdaki kod parçasındaki gibi tanımlanmıştır.

```
function [kenar_ust, kenar_sag, kenar_alt, kenar_sol,
merkez]=odaOlusturma(odaboyu)

% merkezde bulunan ilk blok [0,0] için kenar tanımlamaları

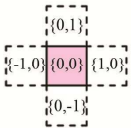
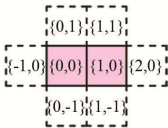
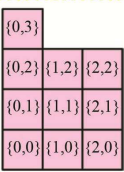
kenar_ust=[0,1,1,1];
kenar_sag=[1,0,1,1];
kenar_alt=[0,0,1,0];
kenar_sol=[0,0,0,1];

for cc=1:odaboyu-1
    tmp=randperm(4);
    tmp_yon=tmp(1);
    ...
    tmp_boy=randperm(tmp);
    tmp_boy=tmp_boy(1);

    % rastgele seçilen bir bloğa rastgele bir yönde blok ekleme
    [kenar_ust kenar_sag kenar_alt kenar_sol]=blokEkle(kenar_ust, kenar_sag,
kenar_alt, kenar_sol,tmp_yon,tmp_boy);
    % kenar çizgilerinden merkez koordinatların çıkarılması
    merkez=kenar2merkez(kenar_ust, kenar_alt);
end
```

Şekil 5.4'te görüldüğü gibi örnek bir kromozomun oluşum adımları fenotipler ile birlikte verilmiştir. İlk adım ilk gen olan $\{0,0\}$ geninin ifade ettiği birime komşu birimlerin genlerinin belirlenmesi ile başlar. Birim karenin dört adet kenarı

olduğundan dört adet komşu kare ve bunlara ilişkin genotipler tanımlanabilir. Bir sonraki aşamada bu dört genden rastgele bir tanesi seçilerek mevcut gen kümesine eklenir. Gen kümesine yeni bir gen eklendiğinde, fenotipte mevcut kenar sayısı 3 artar, dolayısıyla muhtemel komşular kümesine 3 yeni komşu eklenir. Ayrıca yeni eklenen birim muhtemel komşular kümesinden çıkarılır. Özel bir durum olarak yeni eklenen birimin diğer başka bir birime komşu olması durumunun da sorgulanması gerekir. Aksi takdirde bu durum kromozomda aynı genlerin tekrarına neden olur ve kromozom yapısı daha önce verilen eşsiz olma koşulunu sağlamaz. Gen kümesine yeni gen ekleme işlemi, istenen kromozom uzunluğuna ulaşıncaya dek devam eder.

	genotip	fenotip
ilk gen	$\{0,0\}$	
komşu genler	$\{1,0\}, \{0,-1\}, \{-1,0\}, \{0,1\}$	
rastgele seçilen gen	$\{0,0\}, \{1,0\}$	
komşu genler	$\{2,0\}, \{0,-1\}, \{1,-1\}, \{-1,0\}, \{0,1\}, \{1,1\}$	
rastgele seçilen gen	$\{0,0\}, \{1,0\}, \{0,1\}$	
rastgele seçimler devam ediyor	...	...
rastgele seçilen genlerle oluşturulan 10 genli bir kromozom (birey)	$\{0,0\}, \{1,0\}, \{0,1\}, \{0,2\}, \{1,2\}, \{2,2\}, \{1,1\}, \{2,0\}, \{2,1\}, \{0,3\}$	

Şekil 5.4 : Bir oda oluşumunun fenotipler ve genotipleriyle açıklanması.

Bir kromozomun oluşturulması için gerçekleştirilen işlemler ilk popülasyon boyutuna ulaşıncaya kadar tekrar eder. Kromozom oluşumunun her adımı rastgelelik içerdiğinden sonuçta oluşan kromozomlar çeşitlilik gösterir ve yeterince

büyük ilk popülasyon seçimlerinde tüm olası sonuçların elde edilmesi mümkündür. Ancak büyük işlem yükü ve gereksiz sonuçların üretilmesi gibi dezavantajlardan ötürü bu çalışmada, ilk popülasyon boyutu kısıtlı tutularak Genetik Algoritmanın çaprazlama ve uygunluk fonksiyonuna göre değerlendirme aşamalarında çeşitlilik ve iyileşme hedeflenmiştir.

İlk popülasyonun oluşturulmasının ardından bireyler bir uygunluk fonksiyonuna göre değerlendirilir. Kullanılacak uygunluk fonksiyonu matematiksel olarak ifade edilebilir ve genel olarak bir fonksiyonun sahip olması beklenen özellikleri sağlaması gerekir. Kısaca tek bir giriş parametre kümesi için tek bir çıkış üretmeli ve tanım kümesi içinde tüm girişlere yanıt verebilmelidir. Söz konusu oda biçimi üretimi olduğunda uygunluk fonksiyonu olarak çevre/alan oranı kullanılabilir ve uygunluk fonksiyonu şu şekilde tanımlanır:

$$\text{UygunlukFonksiyonu} = \min(\Sigma(\text{BirimKenarSayısı}) / \Sigma(\text{BirimAlanSayısı}))$$

Burada verilen uygunluk fonksiyonu her odayı tanımlayan kromozom için tektir ve değişkenlik göstermez. Popülasyonda bulunan tüm bireylerin dolayısıyla kromozomların uygunluk fonksiyonuna göre değerlendirilmesi ve sıralanması gerekir. Yukarıda verilen uygunluk fonksiyonuna göre bir sıralama söz konusu olduğunda bireylerin en küçük çevre/alan oranına sahip olması hedeflenir. Bu uygunluk fonksiyonunun optimum çözümüne bireylerin şekilleri düzgünleştikçe ve iç açılırları toplamı azaldıkça, bir başka deyişle en/boy oranları 1'e yaklaştıkça ulaşılır. K^2 birim karelik alana sahip bir oda için uygunluk fonksiyonuna göre,

$$\text{Optimum Çözüm} = 4 \times K / K^2 = 4 / K$$

olarak hesaplanır. Ancak Genetik Algoritmanın yapısı gereği amaç optimum çözüme en yakın yoldan ulaşmak değil, optimum çözüme ulaşmaya çalışırken diğer çözümleri de değerlendirmek ve bunların da çözüme katkıda bulunmasını sağlamaktır. Bu özellik gereği algoritmanın adımları ilerledikçe optimum çözüme doğru gidilir iken aykırı çözümler de meydana çıkabilir ve bu da popülasyondaki çeşitliliği artırır. Kromozomların uygunluk fonksiyonuna göre değerlendirilmesi işini gerçekleştiren kod parçası şöyledir:

```
function [y, index]=fitness(populasyon)
len=size(populasyon,1);
y=zeros(len,1);
for i=1:len
```

```

    kromozom=shiftdim(populasyon(i,:,:));
    [k1, k2, k3, k4]=merkez2kenar(kromozom);
    % bir odanın çevresi hesaplanıyor.
    cevre=size(k1,1)+size(k2,1)+size(k3,1)+size(k4,1);
    % bir odanın alanı hesaplanıyor.
    alan=length(kromozom);
    % çevre/alan oranı
    y(i)=cevre/alan;
end
% uygunluk fonksiyonuna göre tüm popülasyonun sıralanması
[y, index]=sort(y);

```

İlk popülasyondaki her bir bireyin uygunluk fonksiyonuna göre değerlendirip derecelendirilmesinin ardından Genetik Algoritmanın elitizm seçim operatörü uygulanır. Bu adımda uygunluk fonksiyonundan en yüksek değeri alan, en uygun bireylerin bir kısmı yeni popülasyona, Genetik Algoritmanın çaprazlama ve mutasyon adımlarına maruz bırakılmadan doğrudan aktarılır. Aktarılacak kromozom sayısı algoritmanın başında tanımlanır ve bu sayı uygulamaya göre değişkenlik gösterir. Elitizm seçim işlemi yoluyla hedeflenen amaç, diğerlerine göre daha iyi bireylerin bozulmadan gelecek nesillere aktarılmasını sağlamaktır. Elitizm işlemini gerçekleştiren kod parçası şu şekildedir:

```

% sıralı popülasyondan ilk kaç tanesinin elit olarak seçildiği bilgisidir.
elit=floor(size(index,1)*elitizm_faktor);

% ilk elit adet kromozom çocuk popülasyonuna katılıyor.
for i=1:elit
    cocuk_populasyon(i,:,:)=populasyon(index(i,:,:));
end

```

Elitizm işleminden arta kalan bireylerden çaprazlama işlemi yoluyla yeni popülasyon üretilir. Az önce de bahsedildiği üzere çocuk popülasyonun ilk birkaç bireyi elitizm seçim yoluyla belirlenirken diğer bireyler çaprazlama sonucu üretilir. Ancak çaprazlama işleminde popülasyon bireyleri seçicilik adımıdaki gibi uygunluk fonksiyonuna göre değil rastgele olarak eşlenirler. Çaprazlama işlemi, bu uygulamaya özgü olarak şu şekilde tekrar yorumlanmıştır. Öncelikle çaprazlanacak aynı boydaki iki kromozomun ortak genleri her iki çocuğa da aynen aktarılır. Birbirinden farklı olan genlerden ise hangilerinin çocuk kromozomlara aktarılacağı rastgele olarak belirlenir. Ancak buradaki kritik nokta, ortak genlerin belirlenmesinin ardından seçilen rastgele genler belirlenen ortak genlere komşu olmayabilir. Bu

durumda şekilsel açıdan bir bütünlük sağlanamaz, üretilen çocuk kromozomların fenotipleri bir odayı ifade edemez. Çaprazlama sonucu ortaya çıkan her kromozomun yine bir odayı ifade edebilmesi için ortak genler belirlendikten sonra bu genlerin komşu genleri hesaplanır. Ardından çaprazlanan iki kromozomdan rastgele biri seçilir ve bu kromozomun rastgele bir geninin komşu genler arasında olup olmadığı kontrol edilir. Eğer bu gen komşu genler arasında değil ise, tekrar bir kromozom rastgele seçilir ve rastgele bir geni belirlenir ve karşılaştırılır. Eğer gen komşu genler arasında ise çocuk kromozoma eklenir. Bu işlem, çocuk kromozom boyu, çaprazlanan kromozomların boyuna eşitlenene kadar devam eder. Çaprazlama işlemini gerçekleştiren fonksiyona ait kod parçası şöyle sunulmuştur:

```
% elitler harici tüm ilk populasyon ikişerli çaprazlamaya giriyor
if rem((popboyu-elit),2)
    ilk=elit;
else
    ilk=elit+1;
end

for i=ilk:2:popboyu-1
    m=0;
    ata1(:,:)=populasyon(i,:,:); % ilk ata
    ata2(:,:)=populasyon(i+1,:,:); % ikinci ata
    % iki atanın kesişen elemanları çocuğa aynen aktarılıyor
    for j=1:odaboyu
        for k=1:odaboyu
            if ata1(j,1)==ata2(k,1) && ata1(j,2)==ata2(k,2)
                m=m+1;
                cocuk_populasyon(i,m,1)=ata1(j,1);
                cocuk_populasyon(i,m,2)=ata1(j,2);
            end
        end
    end

    % kesişen elemanlar iki çocukta da mevcut
    cocuk_populasyon(i+1,:,:)=cocuk_populasyon(i,:,:);
end

% çocukların elemanlarının bir kısmı ataların kesişen kısımlarından
% oluşuyor. Diğer elemanları ise çocuğun mevcut elemanlarına rastgele
% komşular seçilerek bunların hangi ataya ait olduğu kontrol ediliyor
% ve herhangi bir ataya aitse çocuğa ekleniyor. Çocuğun mevcut
% elemanlarından hangisine, hangi yönden komşuluğuna bakılacağı tamamen
% rastsal olduğundan oluşan çocukların iki atadan gelen elemanları
% harici rastlantısal oluşacaktır.
m_global=m;
for c=1:2 % iki çocuk için işlemler tekrarlanıyor
    m=m_global;
    while(m<odaboyu)
        komsu_degil=0;
        tmp=randperm(m);
        secilen_birim=tmp(1); % çocuğa ait kromozomdan rastgele gen
        seciliyor
        tmp=randperm(4);
        secilen_yon=tmp(1); % rastgele bir yön seçiliyor

        % seçilen gen ve yön bilgisine göre komşu elemanın gen bilgisi
        uretiliyor.
```

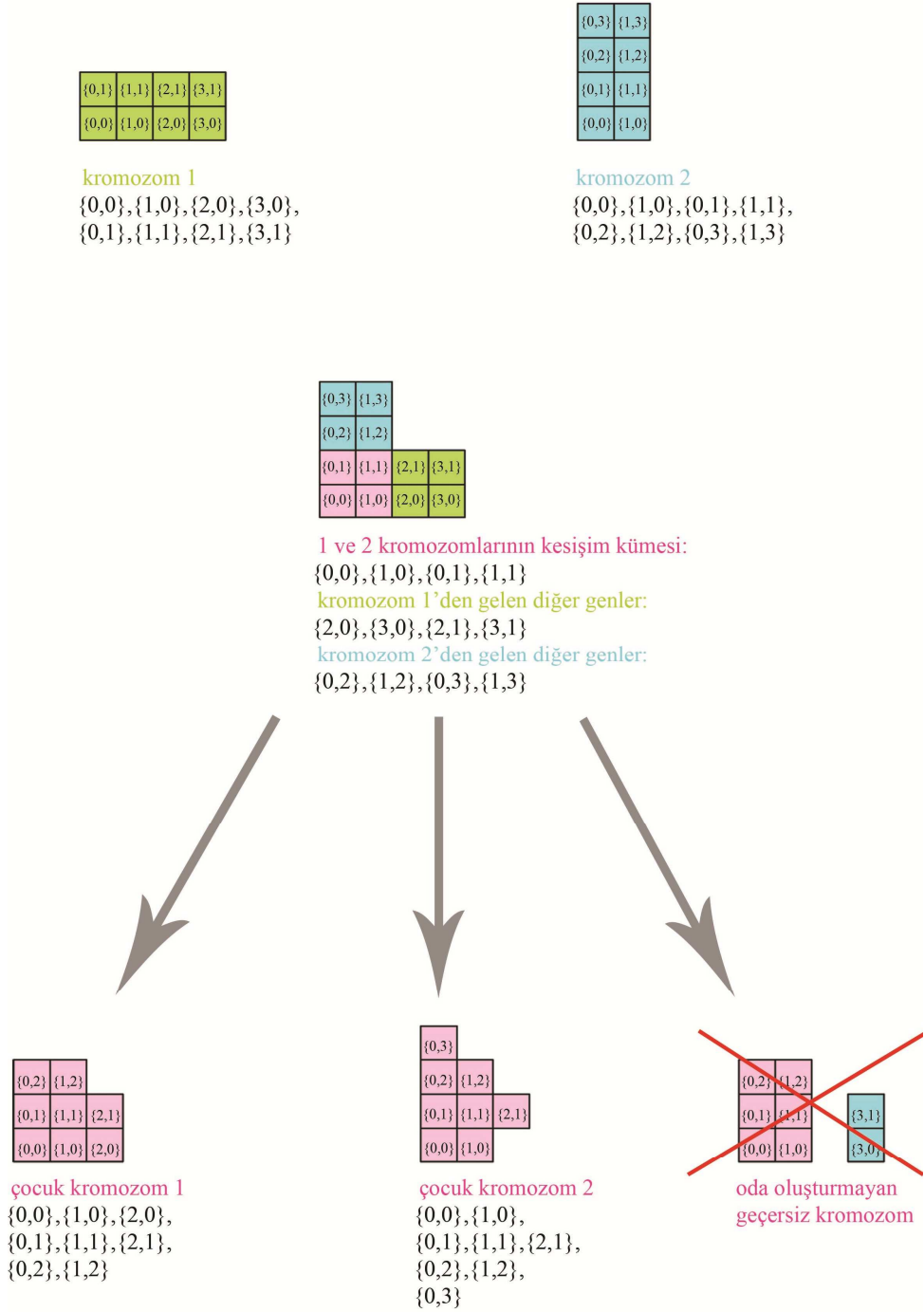
```

        if secilen_yon==1
            komsu=[cocuk_populasyon(i+c-1,secilen_birim,1)
cocuk_populasyon(i+c- 1,secilen_birim,2)+1];
            elseif secilen_yon==2
                komsu=[cocuk_populasyon(i+c-1,secilen_birim,1)+1
cocuk_populasyon(i+c-1,secilen_birim,2)];
            elseif secilen_yon==3
                komsu=[cocuk_populasyon(i+c-1,secilen_birim,1)
cocuk_populasyon(i+c-1,secilen_birim,2)-1];
            elseif secilen_yon==4
                komsu=[cocuk_populasyon(i+c-1,secilen_birim,1)-1
cocuk_populasyon(i+c-1,secilen_birim,2)];
            end

% Belirlenen komşu genin çocuğun bir geni olup olmadığı kontrol
ediliyor.Çocuğun bir %geni ise komşu olamaz, yeni komşu seçmek için döngüden
çıkılır.
        for s=1:m
            if (cocuk_populasyon(i+c-1,s,1)==komsu(1,1) &&
cocuk_populasyon(i+c-1,s,2)==komsu(1,2))
                komsu_degil=1;
                break
            end
        end
        % seçilen gen komşu değil ise bu genin herhangi bir ataya
sahip
        % olup olmadığına bakılır. Hiçbir atada bulunmayan gen
çocukta da
        % bulunamaz. Herhangi bir atada bu gen varsa çocuğa eklenir.
Genin
        % hangi atadan geleceği raststaldır ve çocuğun gen
çeşitliliğini
        % arttırır.
        if not(komsu_degil)
            for j=1:odaboyu
                if (komsu(1,1)==ata1(j,1) && komsu(1,2)==ata1(j,2)) ||
(komsu(1,1)==ata2(j,1) && komsu(1,2)==ata2(j,2))
                    m=m+1;
                    cocuk_populasyon(i+c-1,m,1)=komsu(1,1);
                    cocuk_populasyon(i+c-1,m,2)=komsu(1,2);
                    break
                end
            end
        end
    end
end
end
end

```

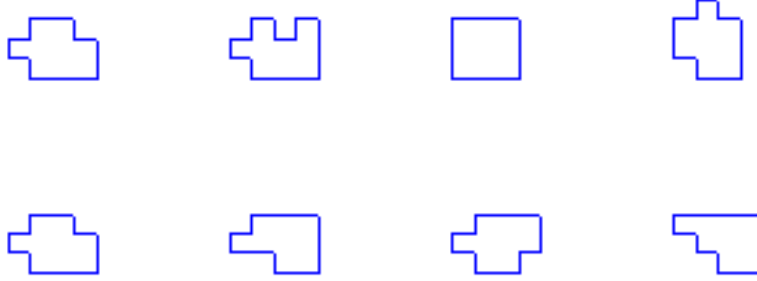
Popülasyona yeni çocuk bireyler üretmek için kullanılan çaprazlama işlemi Şekil 5.5'te bir örnek üzerinden gösterilmiştir.



Şekil 5.5 : Bir çaprazlama işlemcisinin fonksiyonunun oda oluşturma örneği üzerinden gösterimi.

Elitizm ve çaprazlama yoluyla ilk popülasyon kümesinden aynı boyda çocuk popülasyonu üretilmiş olur. Çocuk popülasyonunun üretimi tamamlandıktan sonra popülasyon belirlenen bir kritere göre sınanır ve algoritmanın devam edip etmeyeceğine karar verilir. Bu aşamada kullanılacak ölçüt maksimum döngü sayısına ulaşılması olabileceği gibi, bireylerin uygunluk fonksiyonunun minimum değeri ile olan toplam farkının belirli bir eşik değerinin altına düşmesi şeklinde de

tanımlanabilir. Koşulun sağlanmaması durumunda çocuk popülasyon üzerinden seçicilik adımı ile algoritma devam eder. Şekil 5.6'da algoritmanın ilk birkaç adımı boyunca ortaya çıkan çocuk popülasyonlardan örnekler gösterilmiştir.



Şekil 5.6 : Genetik Algoritmalar ile üretilen 9 birim karelik çocuk popülasyon örnekleri.

Algoritmanın gereği olarak tüm bireyler kısa sürede optimum çözüme ulaşmamakta, her adımda ortaya çıkan çözümler çeşitliliğe neden olmaktadır. Belirlenen sonlanma koşuluna bağlı olarak algoritmanın sonlanması durumunda ortaya çıkan popülasyonda birbiri ile aynı bireylerin ortaya çıkma olasılığı da vardır. Bu yüzden daha sonra kullanılmak üzere veritabanına saklanacak bireyler öncelikle karşılaştırılarak birbirleri ile eş olmamaları sağlanır. Kullanıcı, en yüksek uygunluk değerine sahip, dilediği sayıdaki bireyi veritabanına gönderebileceği gibi, uygunluk değerine bakmaksızın seçtiği herhangi bir bireyi de veritabanında saklayabilir.

5.2.2 Konut kat planı şemasının oluşturulması

Veritabanında saklanan oda şekilleri ve arayüzden girilen kullanıcı tercihleri kullanılarak kat planı şemalarının oluşturulma adımlarını özetleyen akış diyagramı Şekil 5.7'de sunulmuştur. Tümevarım anlayışı ile öncelikle tek tek her odanın biçimleri belirlenmekte daha sonra kat planı üretilmesi için bunlar bir araya gelerek bir bütünü oluşturmaktadır.

Buna göre, veritabanından daha önce kaydedilmiş tüm oda seçeneklerinin yüklenmesinin ardından kat planı üretme algoritması çalıştırılır. Üretilen kat planları, oda yönü tercih kurallarına göre değerlendirilir ve sonuçlar kullanıcıya görsel olarak sunulur. Sonunda da kullanıcının seçtiği planlar veritabanına kaydedilir.



Şekil 5.7 : Odaların oluşturduğu kat planı şemasının akış diyagramı.

Kat planı üretme algoritması veritabanından oda şekillerinin yüklenmesi ile başlar. Yüklenen oda şekilleri veritabanında, daha önce de bahsedildiği gibi odanın alanı uzunluğunda ve odayı oluşturan birim karelerin koordinatlarından oluşan bir dizin şeklinde saklanır. İlk önce tüm odalar, odanın sol alt köşesi [0,0] koordinatından başlayacak şekilde kaydırılır. Ardından kullanıcı tarafından arayüzde girilen arazi ölçüleri bilgisi kullanılarak odaların yerleştirileceği dikdörtgen şeklinde kat alanı oluşturulur. Oluşturulan kat alanı da odalar gibi birim karelerin koordinatları biçiminde tanımlanır.

Bundan sonra bir katta bulunan tüm odalar sırasıyla kat alanı içine yerleştirilir ve diğerleri ile kesişme durumu kontrol edilir. Eğer odalardan herhangi ikisi üst üste çıkıyorsa, bir başka deyişle bazı birim koordinatları aynı ise bu plan geçersiz olur ve tüm odaların sırasıyla x ve y eksenlerinde birer birim kaydırılmış olanları denenir. Bu algoritma tekrarlanarak tüm kat taranana kadar sürer. Algoritmayı gerçekleyen kod parçası şu şekildedir:

```

for i=1:length(odalar)
    s=1;
    for j=1:kat_xlen
        for k=1:kat_ylen
            oda_kay=odalar{i}(:, :) + [(j-1)*ones(size(odalar{i},1),1), (k-1)*ones(size(odalar{i},1),1)];
            if sum(ismember(oda_kay, katmerkez, 'rows'))==size(odalar{i},1)
                plan(s, :, :) = shiftdim(oda_kay);
                s=s+1;
            end
        end
    end
    plan_oda{i}=plan;
    clear plan
end

ana{1}=[0,0];

for i=1: size(plan_oda{1},1)
    ana{i}=shiftdim(plan_oda{1}(i, :, :));
end

for i=1:size(plan_oda,2)-1
    ind=1;
    tmp_total{1}=[0,0];

    for i1=1:size(ana,2)
        for i2=1:size(plan_oda{i+1},1)
            if sum(ismember(ana{i1}(:, :), shiftdim(plan_oda{i+1}(i2, :, :), 'rows'))==0
                tmp_total{ind}=[ana{i1}(:, :); shiftdim(plan_oda{i+1}(i2, :, :))];
                ind=ind+1;
            end
        end
    end
    clear ana

```

```

        ana=tmp_total;
        clear tmp_total
        size(ana)
    end

```

Algoritmanın sonunda ulaşılan “ana” değişkeni, kullanılan oda şekilleri ile mümkün tüm kat planlarını temsil eder. Bu değişkenin boyutu, arazi ölçüleri, odaların ölçüleri, oda şekillerine bağlı olup oda şekillerinin bazı durumlarında ve oda alanları toplamının arazi alanına yaklaştığı durumlarda sıfır olabilir. Diğer tüm durumlarda değişkenin her bir elemanı odaların içerdiği toplam birim kare sayısı uzunluğundadır.

Kullanılan oda şekilleri ile olası tüm kat planlarının üretilmesinin ardından sonuçlar kullanıcı tarafından girilen kısıtlara göre değerlendirilerek bazı sonuçlar elenir. İlk kısıt olan odaların yön koşulu şu şekilde uygulanır: eğer bir odanın kuzeye bakması istenmişse, kat planını oluşturan birim karelerin y ekseninde en büyük değerli olanlarından en az birinin kuzeye bakması istenen odaya ait olması gerekir. Benzer şekilde bir odanın doğuya bakması istenmişse, kat planını oluşturan birim karelerin x ekseninde en büyük değerli olanlarından en az birinin doğuya bakması istenen odaya ait olması gerekir. Güney ve batı koşulu için y ve x eksenlerindeki en küçük değerlere (yani x,y=0) bakılır. Bu aşamayı gerçekleyen kod parçası aşağıda sunulmuştur.

```

for i=1:size(ana,2)
    for j=1:odasayisi
        test_oda=ana{i}( sum(odaboylari(1:j))-odaboylari(j)+1 :
sum(odaboylari(1:j)),1:2 );

        if odayonleri(j)==1 % kuzey koşulu
            if not(ismember(kat_ylen,test_oda(:,2)))
                ana{i}=[0 0];
                break
            end
        elseif odayonleri(j)==2 % doğu koşulu
            if not(ismember(kat_xlen,test_oda(:,1)))
                ana{i}=[0 0];
                break
            end
        elseif odayonleri(j)==3 % güney koşulu
            if not(ismember(0,test_oda(:,2)))
                ana{i}=[0 0];
                break
            end
        elseif odayonleri(j)==4 % batı koşulu
            if not(ismember(0,test_oda(:,1)))
                ana{i}=[0 0];
                break
            end
        end
    end
end

```

```

        end
    end
end
j=1;
for i=1:size(ana,2)
    if not(sum(sum(ana{i}))==0)
        ana_yonlu{j}=ana{i};
        j=j+1;
    end
end
end

```

Kullanıcı tarafından tanımlanan başka kural yoksa algoritma sonunda tüm sonuçlar bir uygunluk fonksiyonuna göre değerlendirilir ve sıralı biçimde kat planı seçenekleri olarak veritabanına kaydedilir. Kullanıcı tarafından bir başka kat planı üzerinde çalışılmak istenmesi durumunda başa dönülerek aynı adımlar tekrarlanabilir. Kullanıcı daha sonra dilerse, sakladığı kat planlarını yükleyerek iki boyutlu ve üç boyutlu olarak görselleştirebilir.

5.3 Modelin Kitlesele Bireyselleştirme Amaçlı Arayüz Tasarımı

Kitlesele bireyselleştirmeyi amaçlayan bu modelin geçerli bir tasarım ortaya koyabilmesi için Genetik Algoritma ile çalışan sistemi veriler ve geri dönüşümlerle destekleyen kullanıcı etkileşimi vazgeçilmez bir unsurdur. Bu etkileşim bilgisayar kullanmayı az bilen kullanıcıya dahi sistemin kullanımını anlatarak ilerleyen etkili arayüzler aracılığıyla sağlanmaktadır. Arayüzlerin akıcılığı ve kullanıcı dostu olması bu tür kullanıcı ile etkileşimli olarak ilerleyen sistemlerde başarılı sonuçlar elde edilmesi açısından büyük önem taşımaktadır.

Modelde kullanılan arayüzler üç ana aşamadan oluşmaktadır. Bunlar, kullanıcıdan bilgi akışının sağlandığı veri arayüzleri, oda biçimi ve kat planı gibi elde edilen tasarımların kullanıcı tarafından değerlendirildiği seçim arayüzleri ve sonuç tasarımın üç boyutlu olarak incelendiği ve veri tabanına kaydedildiği grafik arayüzü olarak tanımlanabilir. Ayrıca her arayüzde bulunan ‘yardımcı paneli’ kullanıcı, program içerisinde arayüzleri nasıl kullanması gerektiğinin bilgilerini vermektedir.

5.3.1 Veri giriş arayüzleri

Veri arayüzleri programın başlangıcında kullanıcıyı karşılayan ve kullanıcıdan tasarımı oluşturmak üzere bilgi alan arayüzlerdir. İlk veri arayüzünde konutun inşa

edileceği arazinin özellikleri ve oluşturulacak konut tipi sorulmaktadır (Şekil 5.8). Kullanıcı ‘arazi özellikleri panelinden’ arazi belirli bir boşluğa oturacak ise sınırlı veya müstakil bir alana oturacaksa sınırsız düğmesini işaretleyerek ebat bilgilerini girmektedir. Daha sonra konutun oturacağı arazi üzerinde yol, manzara gibi herhangi baskın bir yön varsa onun işaretlenmesi gerekmektedir. ‘Konut tipi’ panelinde ise kullanıcıya tasarlamak istediği konut eğer müstakil ise kaç katlı olacağı veya apartman dairesi ise kaçınca katını tasarlamak istediği sorulmaktadır. Gerekli bilgiler verildikten sonra ileri düğmesi ile diğer veri arayüzüne geçilmektedir.

Kullanıcı tarafından, konutta olması istenilen genel oda bilgileri ikinci veri arayüzünde bulunmaktadır. Tümevarım yaklaşımı ile tasarlanan modelde kat planını oluşturacak olan, Şekil 5.9’de görüldüğü üzere bir konutta bulunabilecek sirkülasyon, giriş, oturma odası, mutfak, yemek odası ve yatak odaları, banyolar ve olması istenilen muhtelif odaların alan bilgileri m² üzerinden girilmektedir. Oluşturulmak istenmeyen odalar alan bilgisi verilmeden boş bırakılmaktadır. Buradaki oda kutucuklarındaki alan bilgileri arttıkça, altta bulunan ‘doluluk oranı panelinden’, daha önceki arayüzde girilen konut alanının yüzde üzerinden ne kadar dolduğu anlık veriler ile gösterilmektedir. Eğer istenilirse odaların sabitlemesi istenilen yön bilgisi de girilmektedir. Yönlerin içerisinde dört ana yön bilgisi olduğu gibi daha önceki aşamada girilen manzara da işaretlenebilir. Daha sonra isteğe bağlı olarak odalar arası komşuluk ilişkileri sorulmaktadır. Eğer kullanıcı bir yatak odası ile banyonun yan yana olmasını veya yemek odası ile oturma odasının ve girişin birbirine komşu olmasını istiyorsa bu panelden tıklayarak odaları ilişkilendirebilmektedir. Bu oda bilgilerinin alındığı veri arayüzünde ileri düğmesi işaretlenerek eğer varsa diğer katların verileri de bu şekilde tekrarlanarak, bilgiler belleğe kaydedilmekte ve bir ileri adım olan seçim arayüzlerine geçilmektedir. Eğer arazi bilgileri ve konut tipi ile ilgili değiştirilmek istenilen bir kısım varsa ilk aşamaya da geri dönebilir.

Konut Planı Oluşturma Asistanı

KİTLESEL BİREYSELLEŞTİRME AMAÇLI KONUT TASARIMI

Arazi Özellikleri

☒ Sınırlı ☐ Sınırsız

En: m Alan: m²

Boy: m

Manzara Yönü:

Konut Tipi

☐ Müstakil:

☒ Apartman Dairesi:

Yardımcı

Arazi özellikleri panelinden konutun yerleşeceği arazi sınırlı ise en/boy değerlerini, sınırlı değil ise alanı giriniz.

Arazi, konutun oda yerleşimini etkileyebilecek bir manzaraya sahip ise yönünü seçiniz.

Konut tipi panelinden ilgili konut tipini ve müstakil bina için kat sayısını, apartman daireleri için dilediğiniz kat numarasını seçiniz.

İLERİ

Şekil 5.8 : Arayüz üzerinden modele arazi özellikleri ve konut tipi bilgilerinin girilmesi.

Konut Planı Oluşturma Asistanı

KİTLESEL BİREYSELLEŞTİRME AMAÇLI KONUT TASARIMI

Konut Tipi: **Müstakil -1**

Oda Ölçüleri

	Alan (m ²)	Yön	Komşuluk ilişkisi
Sirkülasyon:	8	Farketmez	☐
Giriş:	0	Farketmez	☐
Oturma Odası:	30	Farketmez	☐
Mutfak:	0	Farketmez	☐
Yemek Odası:	0	Farketmez	☐
Yatak Odası-1:	30	Farketmez	☐
Yatak Odası-2:	0	Farketmez	☐
Banyo-1:	20	Farketmez	☐
Banyo-2:	0	Farketmez	☐
Oda:	0	Farketmez	☐
Oda:	0	Farketmez	☐

Doluluk Oranı: **59 %**

Yardımcı

Konut tipi özelliği bir önceki arayüzden otomatik olarak ayarlanır. Oda Ölçüleri panelinden Konut Tipi özelliğinde gösterilen kat için oda ölçülerini giriniz ve terah edilen oda yönünü seçiniz.

Odalar arası komşuluk ilişkisi tanımlamak isterseniz kutuya tıklayınız.

İstemediğiniz oda için alan kutusunu boş bırakınız. Panelin altındaki araç kat alanının yüzde kaçının kullanıldığını gösterir.

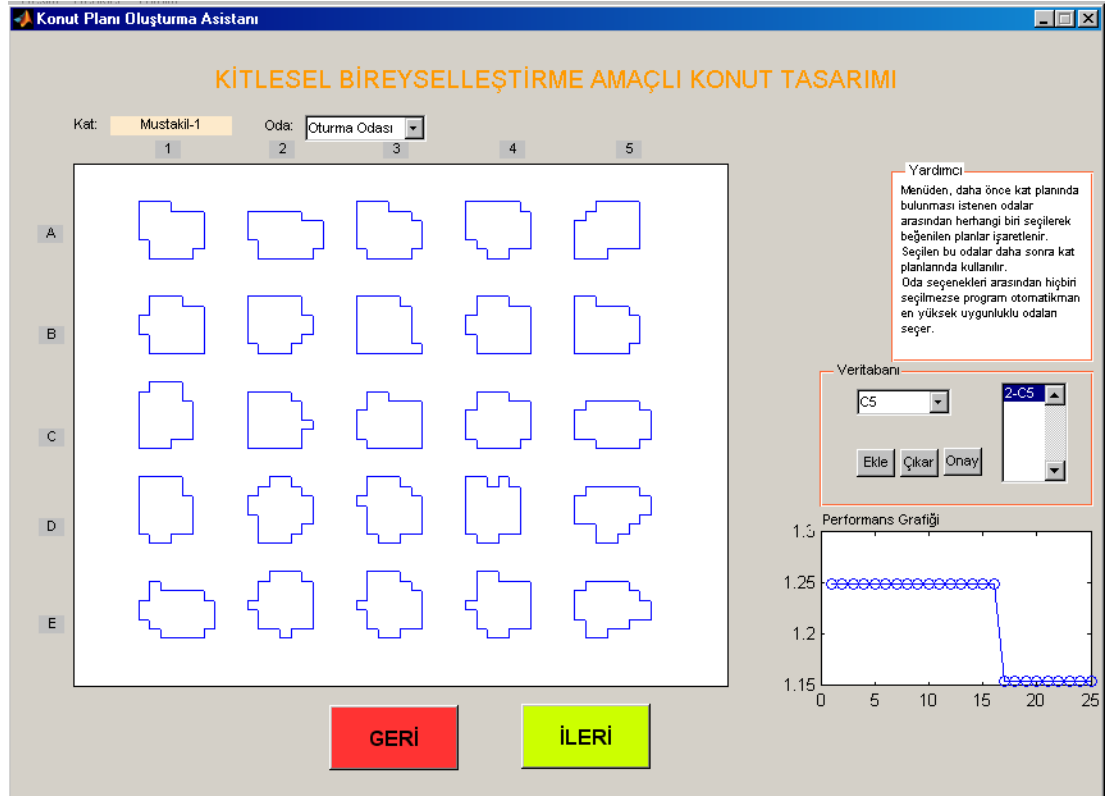
Bu sayfayı tamamladığınızda ileri tuşuna basınız.

GERİ **İLERİ**

Şekil 5.9 : Arayüz üzerinden modele odaların olması istenilen topolojik özelliklerin girilmesi.

5.3.2 Seçim arayüzleri

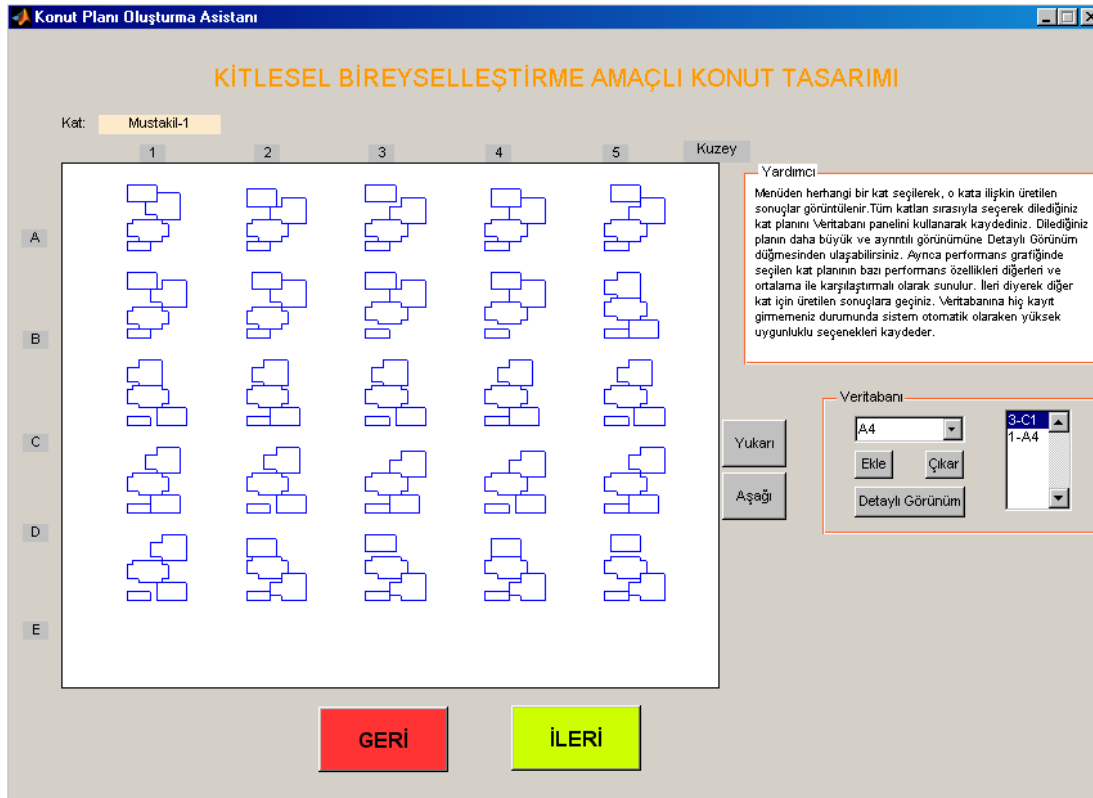
Seçim arayüzlerinden ilki olan odaların biçimlerinin seçiminin yapıldığı arayüz üzerinden, her odanın Genetik Algoritma ile işlemcileri aracılığı ile üretilen uygunluk fonksiyonunu sağlayan en iyi sonuç biçimleri kullanıcıya görüntüleme paneli üzerinden sunulmaktadır. Şekil 5.10'da görüldüğü üzere kullanıcı, popülasyondaki evrimleşme sonucu üretilen bireylerin her birinin performans grafiği üzerindeki yerlerini de kaliteleri açısından tespit ederek, biçim olarak beğendiklerini kod numaralarıyla veritabanı paneli üzerinden ekleyebilmektedir. Eğer sonradan çıkarmayı düşündüğü oda biçimleri olursa bunları tekrar aynı panel üzerinden çıkarabilmektedir. Şekil 5.11'de görüldüğü gibi istenilen her bir oda için oda şekillerinin seçimi kullanıcı tarafından tamamlandıktan sonra diğer seçim arayüzüne geçilir.



Şekil 5.10 : Oda tasarımların arayüz üzerinden kontrol edilerek kullanıcının beğenisine göre seçimlerin yapılması.

Bir diğer seçim arayüzü ise kat planlarının oluşturulmasını sağlayan, Şekil 5.11'de görülen kat planı seçim arayüzüdür. Burada ana panel üzerinde kod numaraları ile görselleştirilen ve bir önceki aşamada seçilen oda biçimleri ile oluşturulmuş ve kullanıcının modelin veri arayüzünde belirttiği yön, manzara kısıtlamala bilgilerine

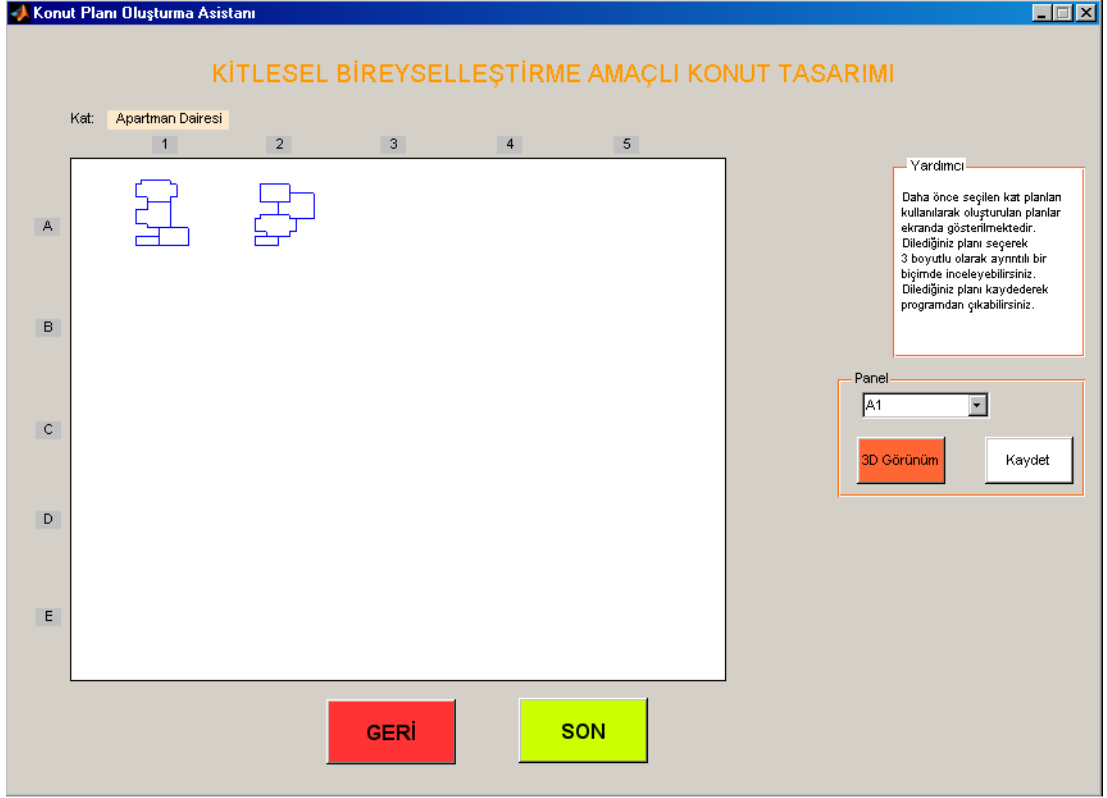
göre bir araya getirilmiş olan odalar veritabanı paneli üzerinden seçilmektedir. Veritabanı panelinde bulunan ‘detaylı görünüm’ düğmesi üzerinden, kat planı iki boyutlu olarak daha büyük imaj ile görüntülenebilmektedir. Diğer seçim arayüzünde olduğu gibi performans grafiği panelinden kat planlarının uygunluklarına göre nasıl bir eğri oluşturdukları görülmektedir. Tercihler kullanıcının kendi istek ve zevkine bırakıldığı için buradaki seçim tamamı ile kitlesel bireyselleştirme mantığına uygundur.



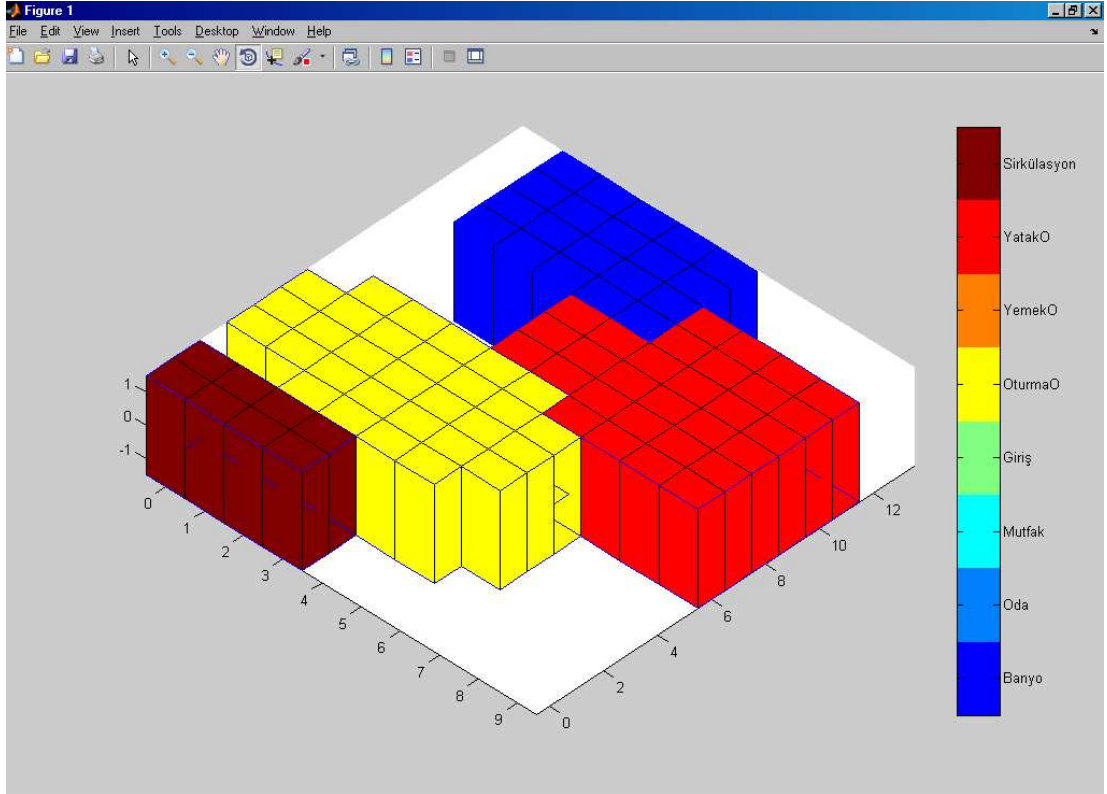
Şekil 5.11 : Oluşturulan kat planlarını arasından tercih yapılan seçim arayüzü.

5.3.3 Grafik arayüzü

Üretilen kat planlarını ve oluşturulan konutun formunu görüntülemeyi gerçekleştirmek için grafik arayüzleri kullanılmaktadır. Şekil 5.12’de bir grafik arayüz penceresi görülmektedir. Bu arayüzde kullanıcının seçtiği kat planı şemaları iki boyutlu olarak gözlemlenebilmektedir. Kullanıcı bu kat planlarından birini daha detaylı incelemek isterse kullanıcı paneli üzerinden seçmektedir.



Şekil 5.12 : Tercih edilen kat planı tasarımını üç boyutlu olarak görselleştiren grafik arayüzü.

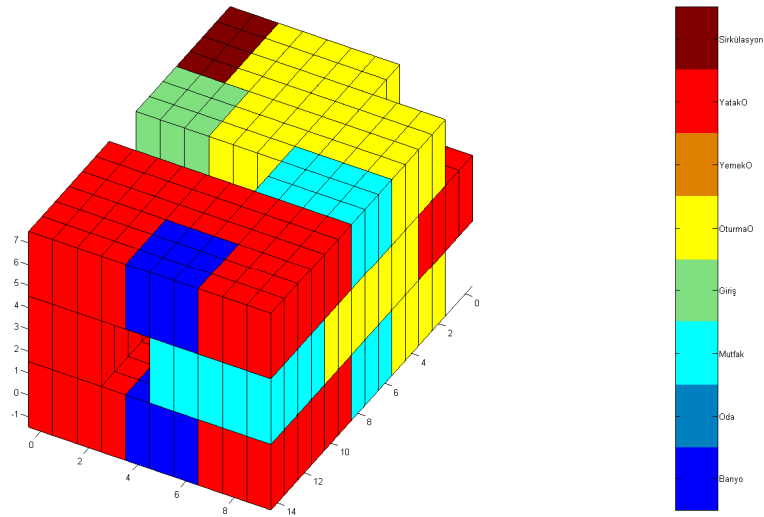


Şekil 5.13 : Üç boyutlu görselleştirmeyi sağlayan grafik arayüzü.

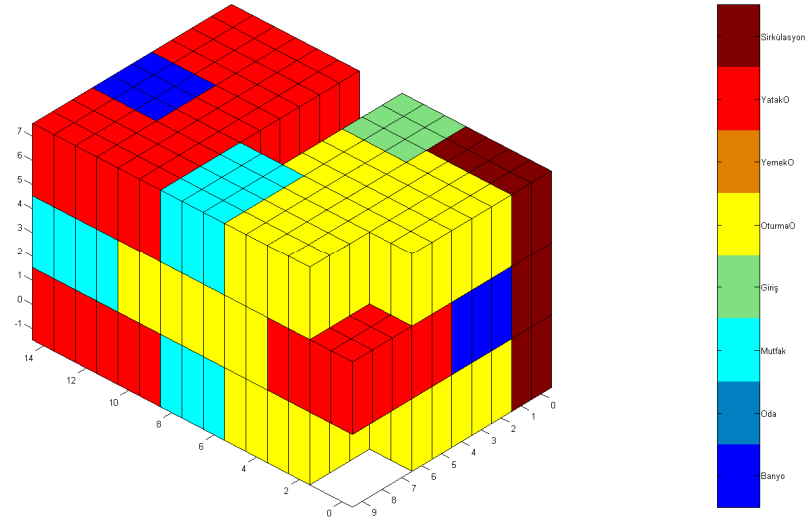
Kullanıcı, seçilen kat planını aynı zamanda üç boyutlu olarak daha iyi anlamak açısından, aktif bir şekilde fare yardımı ile yaklaşma, uzaklaşma, döndürme gibi hareketlerle (Şekil 5.13) izleyebilmektedir. Üç boyutlu grafik arayüzünde odalar yan panelde verilen bir renk skalası ile ayrılmaktadır. En son aşama olarak, kullanıcı üç boyutlu olarak da deneyimlediği kat planlarından en uygun bir tanesini seçerek üretilebilmesi amacıyla mimara aktarılması için veritabanına kaydetmektedir.

5.4 Farklı Kullanıcılara Özgü Konut Tiplerinin Geliştirilmesi

Genetik Algoritmaya dayalı kitlesel bireyselleştirme amaçlı konut tasarım modeli ile farklı arazi üzerine kurulacak, farklı kişi sayısına uygun, farklı konut türleri tasarlanabilir. Bu farklı konut tiplerinden biri için, arazi sınırları belirli olan, 3 katlı ve çok çocuklu bir ailenin tasarladığı bir konut örnek olarak verilebilir. Şekil 5.14 ve Şekil 5.15’de görüldüğü üzere belirli arazi sınırları üzerine kurulmuş, ıslak hacimleri yatak odaları ile ilişkilendirilmiş, her katta bir yaşam alanı olan müstakil bir konut tasarlanmıştır. Alandaki her birimkare, bir metrekareye denk gelmektedir. Binadaki düşey sirkülasyon, her katta sabit bir alana yerleştirilmiştir. Algoritmanın çalışması sonucunda elde edilen bina kütesinde, açıkta kalan mekanlar hol, balkon, teras, dolap, havalandırma ve tesisat shaftı boşluğu olarak değerlendirilebilir.

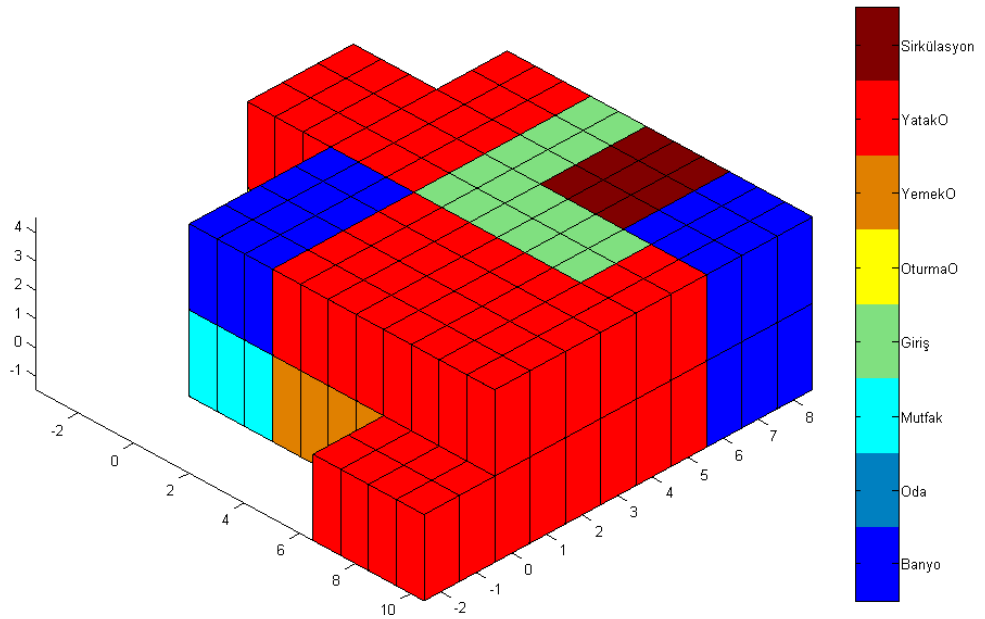


Şekil 5.14 : Bir aileye ait, üç katlı, sınırları belirli konut tasarımı perspektif 1.

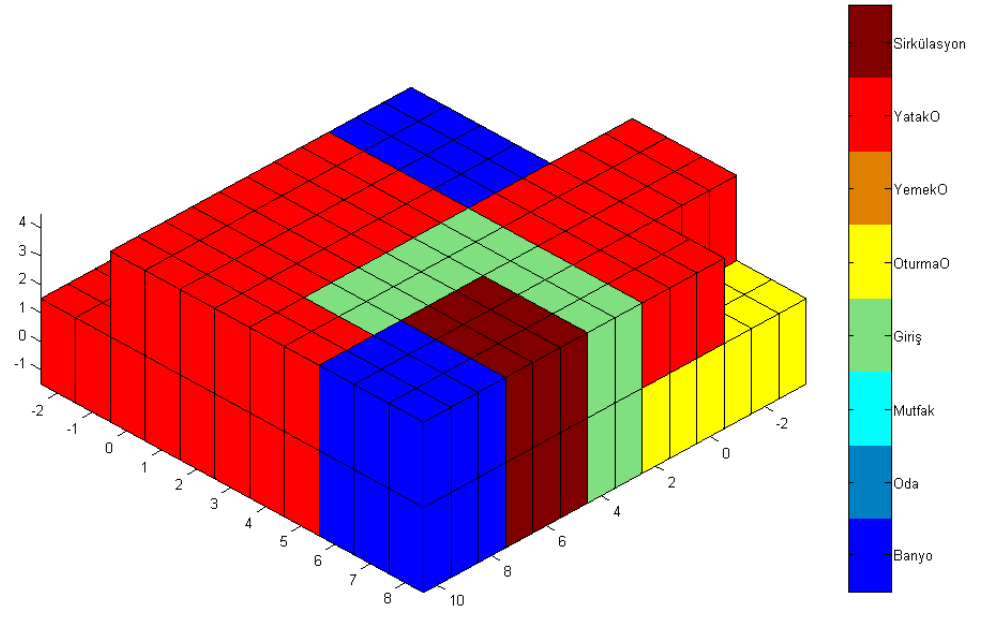


Şekil 5.15 : Bir aileye ait, üç katlı, sınırları belirli konut tasarımı perspektif 2.

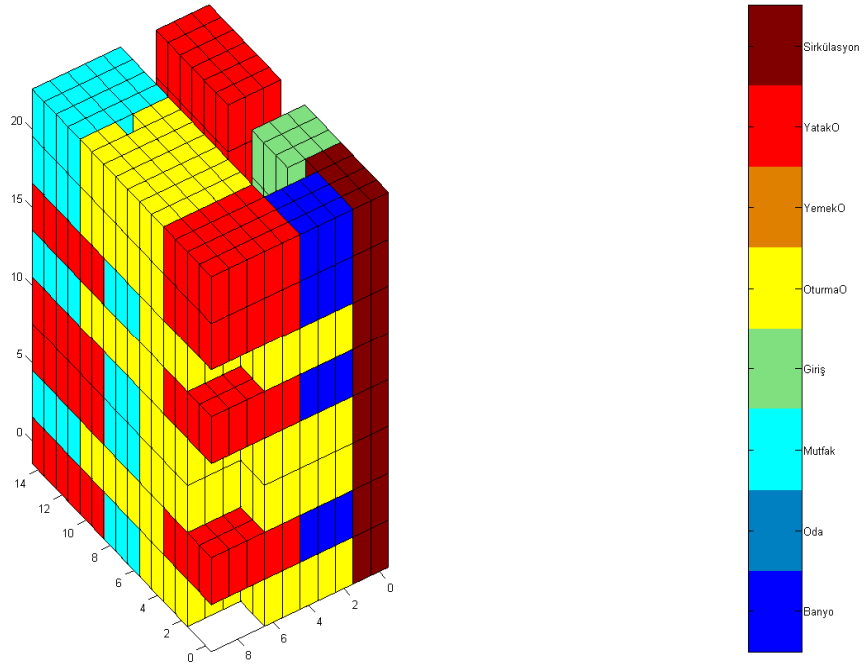
Genetik Algoritma kullanılarak elde edilen bir başka konut ise müstakil bir alan üzerinde, katı arazi kısıtlamaları olmadan tasarlanan iki katlı bahçe içerisinde yazlık kullanıma uygun bir bina olabilir. Bu durumda kullanıcı, veri giriş arayüzü üzerinde arazi seçeneklerinden sınırsız olanı ve konut tipleri içerisinde de müstakil konut tipini seçmelidir. Zemin katta genel kullanım alanlarından sirkülasyon, giriş, banyo, mutfak, yemek odası varken bir adet misafir yatak odası bulunabilir. Birinci katta ise sadece konut kullanıcılarının kullanacağı özel yaşam alanları vardır (Şekil 5.16; Şekil 5.17).



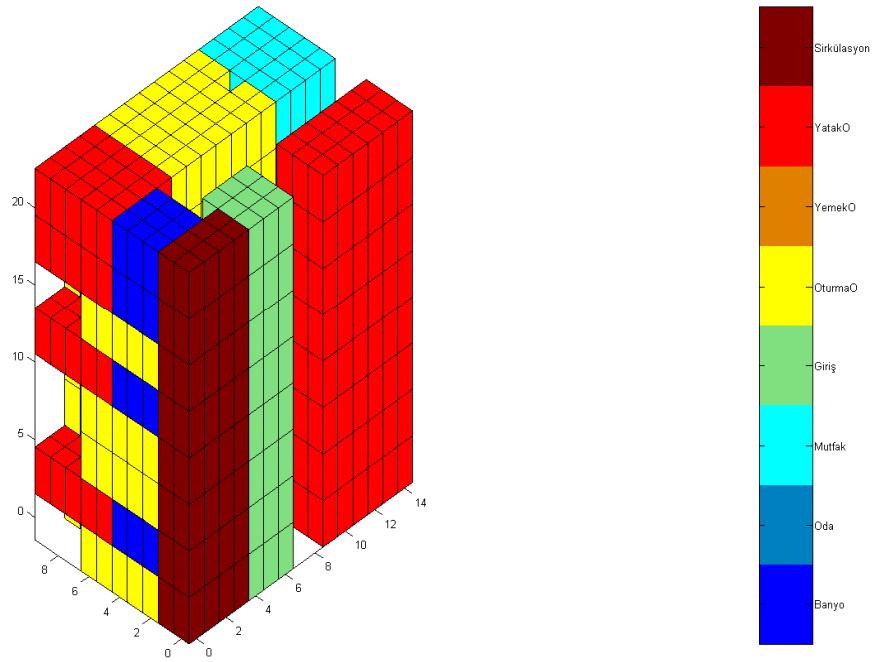
Şekil 5.16 : İki katlı, sınırları belirsiz, yazlık konut tasarımı perspektif 1.



Şekil 5.17 : İki katlı, sınırları belirsiz, yazlık konut tasarımı perspektif 2.



Şekil 5.18 : Farklı kullanıcılara ait çok katlı konut tasarımı perspektif 1.



Şekil 5.19 : Farklı kullanıcılara ait çok katlı konut tasarımı perspektif 2.

Farklı bir senaryo olarak, her kullanıcının hangi dairede oturacağını bilindiği çok katlı bir konut yapısında, her dairenin kullanıcılarının kendi konutunu tasarlaması ve veritabanına kaydetmesi sonucu Şekil 5.17'deki gibi bir konut formu ortaya çıkabilir. Binaya ait girişin ve dikey sirkülasyonun yeri sabit bırakılarak, her katta farklı kullanıcılar kendi kişisel beğeni ve ihtiyaçları doğrultusunda kendi kat planlarını oluşturabilirler (Şekil 5.18). Bir kat planına ait tüm bilgiler veritabanına kaydedilmektedir, bunun sonucunda kullanıcılar başka komşu kullanıcıların kat tasarımlarını beğenerek bu tasarımları değiştirmeden veya ufak değişikliklerle kendi katları için de uygulatabilirler. Kullanıcıların gereksinimleri ön planda tutularak, işlevin bir bina formunu oluşturması konut kitlesine hareket katabilir.

Tez kapsamında, yukarıda açıklanan uygulama örneklerinde de görüldüğü gibi, genetik algortimaların doğal evrimleşme sürecini taklit ederek geliştirilen, seçme, çaprazlama, mutasyon gibi işlemcileri kullanarak ve kullanıcı isteklerini de ön planda tutarak, kullanıcının aktif olarak tasarım sürecinde yer aldığı bir konut tasarımı modeli geliştirilmiştir. Böyle bir modelin kendi müstakil evini oluşturmak isteyen kullanıcıların tasarım sürecine katılımlarında veya farklı yapılarıdaki ailelerin oturduğu çok katlı konutların tasarımında, konut kullanıcılarının isteklerinin iki ve üç boyutlu görüntüleri ile mimara aktarılması aşamasında yardımcı bir araç olarak kullanılması düşünülmüştür.

6. SONUÇLAR

Bilgisayar destekli bilgi sistemlerinin kontrolünde farklı üretim çeşitleri ile her bir müşteri için özelleşmiş ürünün tasarlanıp üretilmesine kitlesel bireyselleştirme adı verilmektedir. Farklı bir deyiş ile kitlesel bireyselleştirme, seri üretim ile bireyselleştirmenin en iyi yönlerinin bir araya getirilerek birleştirilmesidir (Yolovich, 1993). Farklı iş sahalarında kitlesel bireyselleştirme uygulamaları ile kullanıcıların kişisel ihtiyaçlarına ve isteklerine yönelik, üretim sistemlerindeki bilgisayar teknolojileri kullanılarak hızlı üretim yapılabilmektedir. Mimari konut tasarımında da kitlesel bireyselleştirme, günümüzdeki yapı sektöründe sadece malzeme seçimleriyle kısıtlı da olsa kullanılmaya başlanmıştır. Seri üretim anlayışı ile her türden kullanıcı için aynı konut tiplerinin firmalar tarafından veya genel anket sonuçları ile belirlendiği konut tasarımları kullanışlılık, estetik ve maliyet açısından kullanıcıları eşit derecede tatmin edememektedir. Bu sebeple üreticilerin (buradaki üretici mimar anlamına da gelebilir) karar verip müşterilerin razı olduğu eski anlayış, kitlesel bireyselleşme ile gelişmekte ve müşteri ile üreticinin beraber karar verip uyguladığı veya müşterinin karar verip üreticinin uyguladığı bir yaklaşıma dönüşmüştür. Bu bağlamda konut kullanıcılarının kendi yaşamak isteyecekleri yapının temel gereksinimlerini ve biçimini en iyi kendilerinin söz sahibi olacağı bir tasarımla oluşturabileceği düşüncesi bu tez çalışmasının da ana fikrini oluşturmaktadır. Bilgisayar destekli model üzerinden üretilen çalışmada, kat planı oluşturma ve form arama sistemi Genetik Algoritmaların kullanıldığı evrimsel bilgi işleme ile yürütülmektedir.

Evrimsel bilgi işleme, son on beş yılı aşkın süredir tasarımlar üretmek amacıyla farklı biçimlerde karşımıza çıkmaktadır. Tasarımcılar tasarımlarının belirli bir kısmını en iyilemek, sanatçılar estetik olarak daha memnuniyet verici formlar yakalamak, mimarlar yeni bina tasarımlarını eskiz aşamasından evrimsel gelişime tabi tutarak form üretmek amacıyla ve bilgisayar bilimcileri yapay hayatın kontrol sistemlerini geliştirmek için evrimi kullanmaktadırlar (Bentley, 1999). Evrimsel Algoritmalar ile elde edilen tasarım sistemleri günümüzde yoğun olarak kullanılan

bilgisayar destekli tasarım yaklaşımlarından biridir. Doğadan esinlenilerek geliştirilen bu yaklaşımlar evrim kurallarını, popülasyondaki bireylerin hayatta kalma çabasını ve çevresel kısıtlamaları taklit ederek bilgisayar ortamına dönüştürmeyi amaçlamaktadır. Bu doğal sürecin bilgisayar sistemlerine adapte edilmesi ile bir problemin çözümündeki en iyileme yaklaşımının kısa sürede düzenlenmesi ve evrimleşen sonuçlar ile yaratıcı ve beklenmedik sonuçlar elde edilmesi hedeflenmektedir.

Bilgisayar destekli evrimsel algoritmaların, bu tez çalışması için tasarlanan modeldeki gibi kat planı oluşturma ve form arama çalışmalarında kullanılması daha önceki başlıklardaki örneklerde de anlatıldığı üzere yeni bir kavram değildir. Bu çalışma mimar veya bilgisayar programcısı olmayan gelecekteki konut kullanıcılarının kişisel beğeni ve istekleri veri girişine dönüştürölüp ön planda tutularak, bilişimdeki daha önce yapılan çalışmaların ışığında Genetik Algoritma kavramı kullanılarak üretilmiştir. Tümevarım yaklaşımı ile model süresince kullanıcılardan öncelikle oluşturulacak mekanların boyut ve kısıtlamaları istenmektedir; üretilen farklı sonuçlar kullanıcıya sunularak seçimi beklenmekte ve böylece interaktif bir tasarım geliştirilmektedir. Yani tasarımların ana karar verme mekanizması kullanıcının kontrolüne bırakılmıştır, fakat bir sonuca gidilirken modelin kullanıcıyı yönlendirmesi de sürecin hızlanmasına katkı sağlamaktadır. Binlerce farklı tasarım arasından kısıtlara uygun seçim ve en iyi sonuçlar içinde tekrar üretim Genetik Algoritma ile sağlanmaktadır. Bütün tasarımlar için böyle çok yönlü bir değerlendirmeyi, bir tasarımcının geleneksel eskiz yöntemleri ile yapabilmesi olanaklı değildir. Eskiz yöntemi ile çalışan bir tasarımcının ihtimal vermeyeceği farklılıkta sürpriz formlar veya kat planları model sonucu olarak üretilebilmekte ve model kullanıcı verileri ile yapılandırıldığı için bu sonucun aynı zamanda kullanıcının tercihleri ile de örtüşmesi sağlanmaktadır. Bu nedenle elde edilen evrimleşmiş, farklı ve tatmin edici mimari tasarım sonuçları, uzman tasarımcı olmayan kullanıcı ile mimar arasında bir bağlantı kurarak tasarım aşamasını kısaltıp konutun üretim aşamasına daha hızlı geçişine yardımcı olabilir.

Tasarlanan modeli daha önce yapılan evrimsel algoritmalar kullanılarak elde edilen çalışmalar ile karşılaştırmak gerekirse öncelikle bu çalışmanın bilinçli olarak kullanıcı odaklı, yani kitlesel bireyselleştirme amaçlı yapıldığının altının çizilmesi gerekmektedir. Elezkurtaj, Rosenman ve Chouchoulas'ın tasarımlarında arayüz

üzerinden programın yürütülmesini gerçekleştiren uzman tasarımcı olan mimardır (Elezkurtaj & Franck, 1999; Rosenman, 1996; Chouchoulas, 2003). Bu çalışmada ise, konutu kullanacak olan ve uzman tasarımcı olmayan kullanıcının kendi istek ve düşüncelerini arayüze veri olarak girmesi ve etkileşimli bir şekilde ilerleyip taslak olarak iki ve üç boyutlu konut tasarımı elde etmesi beklenmektedir. Örneğin, Rosenman (1996) tasarımında profesyonel tasarımcıya yönelik bir program geliştirdiği için GA parametreleri kullanıcıdan bir girdi olarak alınıyor. fakat bu tez çalışması kapsamında geliştirilen modelde ise kullanıcı uzman olmadığından dolayı belirli GA parametreleri önceden belirlenerek kullanıcıya hazır olarak sunulmaktadır. Arayüzde sağlıklı sonuç alınabilmesi için her adımda yardımcı bilgiler olması da modelin kullanıcı dostu olduğu fikrini desteklemektedir.

Tomor Elezkurtaj'ın tasarladığı kat planı üretim modelinde, evrimleşme süreci sınırlı dörtgenlerden oluşan tek katlı ve iki boyutlu bir çerçeve içerisinde tamamlanmaktadır (Elezkurtaj & Franck, 1999). Bu tasarımı bir noktadan sonra tekdüzeleştirmekte ve olası hareketli sonuçlar elde edilmesi engellenmektedir. Rosenman'ın oluşturduğu iki boyutlu ve tek katlı konut planları tasarım modelinde ise tümevarım yöntemi ile GA'lar kullanılarak oluşturulan farklı şekillerdeki oda birimleri bölgeleri, bölgeler de birleşerek kat planlarını üretmektedir ve tasarım çok kenarlı odalardan bir araya gelerek sözü edilen monotonluktan sıyrılmaktadır (Rosenman, 1996). Fakat burada da tasarımda üretilen sonuçlar sadece tek kat ile sınırlı kalmıştır. Tez çalışmasındaki modelde daha önce yapılan çalışmalarda bu biçim sorunları tespit edilerek, karşılaşılan problemlerin nasıl aşılabileceği üzerine yoğunlaşmıştır. Tümdengelim yaklaşımı ile bir çerçeve içerisine doğrudan odaları sığdırmaya çalışmak yerine, ilk önce en küçük birim olan odaların tek tek GA'lar ile üretilmesi ve bunların belirli kurallar içerisinde bir araya gelerek kat planını oluşturması tasarlanmıştır. Fakat bu yapılırken iki olasılıklı olarak odaların kullanıcının arayüz aracılığı ile belirttiği sınırlı bir alanda bir araya gelmesi veya sınırsız bir boşlukta odaların ilişki kurması sağlanmıştır. Böylece kullanıcının konut arazisinin tercihinine göre, hem sınırlı bir alan içerisine odaların birbirinden değişik biçimleri sebebiyle farklı boşluklar oluşturularak plan şeması hareketlenmektedir, hem de sınırsız alan seçeneğinde birden çok katlı konut tasarımları sirkülasyon boşluğu sabit tutularak belirli sınırlandırmalar ile üretilebilmektedir. Bu çok katlı üretim aynı zamanda sınırlı alan seçeneğinde de gerçekleştirilebilmektedir.

Tasarlanan model çerçevesinde üretilen sonuçlar şu an için sadece dik açılı biçimler ile sınırlı kalmaktadır. İleride 90 dereceden farklı açılar, daireler veya eğrisellikler de kullanılabilir. Arazi bilgileri de en boy bilgileri girişi ile sınırlı kalmayıp, çok çeşitli şekillerde olan alanların koordinatları girilerek arazi yapısına uygun tasarımlar geliştirilebilir. Bu hem araziye net olarak kullanabilme hem de farklı geometrik tasarımların üretim olanağını sağlar.

Ayrıca geleceğe yönelik hedeflerden bir tanesi de MATLAB kullanılarak tasarlanan modelin sonuç üretimlerinin, bir ara programı ile iki boyutlu ve üç boyutlu mimari yazılım programlarında kullanabilmek için dönüştürülmesidir. Böylece kullanıcının istekleri net olarak firma sahibine veya mimara iletilebilecek ve oluşturulan tasarım üzerinden projelendirmeye doğrudan geçiş imkanı olabilecektir.

Sonuç olarak, insanların en temel ihtiyaçlarını içerisinde yaşayarak giderdiği konutun tasarımı ve üretimi, ister geleneksel eskiz yöntemleri ile, ister bilgisayar teknolojileri ile yapılacak olsun, kullanışlılık ve kullanıcı memnuniyeti açısından içerisinde yaşayacak olan insanların fikri ve estetik görüşleri de ön planda tutularak gerçekleştirilmelidir. Kitlesel bireyselleştirme odaklı konut tasarımı amacıyla tez çalışması kapsamında Genetik Algoritmalar kullanılarak geliştirilen model gibi yeni konut tasarım araçlarıyla, böyle tasarım problemlerinin büyük ölçüde bilgisayar destekli mimari tasarım ile çözülebileceği ortaya konulmuştur. Bu tür sistemlerin tasarımda kullanılmasının sadece araç olarak algılanması süreci yavaş yavaş yerini sistemlerin tasarımı gerçekleştiren yaratıcı olarak görülmesine bırakmaktadır ve böylece bilgisayar destekli mimari tasarım sistemleri zamanla daha sık kullanılacaktır ve gelişimi ilerletilecektir.

KAYNAKLAR

- Åhlström, P. & Westbrook, R.**, 1999. "Implications of Mass Customization for Operations Management", *International Journal of Operations & Production Management*, **19-3**, 262-274.
- Arief, A. & Burkhart, B.**, 2002. *Prefab*, Gibbs Smith, Utah.
- Bardakçı, A.**, 2004. Kitlesel Bireyselleştirme Uygulama Yöntemleri, *Akdeniz İ.İ.B.F. Dergisi*, **8**,1-17.
- Bedir, A.**, 2002. Türkiye’de Otomotiv Sanayi Gelişme Perspektifi, *Devlet Planlama Teşkilatı*, Yayın No 2660.
- Bentley, P.**, 1999. "An Introduction to Evolutionary Design by Computers", *Evolutionary Design by Computers*, Morgan Kaufmann Publishers Inc., San Francisco.
- Bi, Z.M. & Zhang, W.J.**, 2001. Modularity Technology in Manufacturing: Taxonomy and Issues, *The International Journal of Advanced Manufacturing Technology*, **18-5**, 381-390.
- Boden, M. A.**, 1990. The creative mind: Myths and mechanisms. London: Weidenfeld & Nicolson.
- Boër, C. & Dulio, S.**, 2007. Mass Customization and Footwear: Myth, Salvation or Reality?, London, Springer.
- Bolat, B., Erol, K., ve İmrak, C.**, 2004. Genetic Algorithms in Engineering Applications and the Function of Operators, *Journal of Engineering and Natural Sciences*, 264-265.
- Bulca, A.**, 1977. İzmit Yeni Yerleşimler Projesi, *Mimarlık*, **151**.
- Chen, C.L., Vempati, V.S., Aljaber, N.**, 1995. An Application of Genetic Algorithms for Flowshop Problems, *European Journal of Operational Research*, **80**, 389-396.
- Chouchoulas, O.**, 2003. Shape Evolution: An Algorithmic Method for Conceptual Architectural Design Combining Shape Grammars and Genetic Algorithms, *PhD Thesis*, University of Bath.
- Clark, E. Jr.**, 1986. The American Family Home, MIT Press, Cambridge, MA.
- Cliff, C., Husbands, P., Meyer, J., Wilson, S.W.**, 1994. From Animals to Animats 3. Proceedings Of The Third International Conference On Simulation of Adaptive Behaviour, MIT Press.
- Cuperus, Y.**, 2003. Mass Customization in Housing an Open Building /Lean Construction Study, *Dense Living Urban Structures International Conference on Open Building*, Hong Kong.
- Çavdar, T.**, 1978. Toplum Bilinçlenmesinde Araç Olarak Kalıtsal Tasarım: İzmit Yenilikçi Yerleşmeler Projesi, *Mimarlık*, **154**.

- Davis, L.**, 1991. The Handbook of Genetic Algorithms, Van Nostrand Reinhold, New York.
- Davis, S.**, 1987. Future Perfect, Addison-Wesley, New York.
- Dawkins, R.**, 1986. The Blind Watchmaker, Longman Scientific & Technical Pub.
- Duarte, J. P.**, 2005. A discursive grammar for customizing mass housing: the case of Siza's houses at Malagueira, Automation in Construction, **14**, 265-275.
- Eksin, I. & Erol, O.K.**, 2001. Evolutionary Algorithm with Modifications in the Reproduction Phase, *IEE Proceedings-Software*, **148-2**, 75-80.
- Emel , G.G. & Taşkın Ç.**, 2002. Genetik Algoritmalar ve Uygulama Alanları, *Uludağ Üniversitesi İktisadi ve İdari Bilimler Dergisi*, **XXI-I**, 129-152.
- Fogel, L. J., Owens A. J. ve Walsh, M. J.**, 1966. Artificial Intelligence through Simulated Evolution, Wiley, New York.
- Gero, J. S.**,1996. Computers and Creative Design, In M.Tan and R.Teh (eds), The Global Design Studio, National University of Singapore, 11-19.
- Gilbert, A.**, 1994. The Latin American City, Russel Press, Nottingham.
- Gilmore, J. H. & Pine, B. J.**, 1997. The Four Faces of Mass Customization, *Harward Business Review*, **Jan-Feb**, 91-101.
- Goldberg, D. E.**, 1989. Genetic Algorithms in Search, Optimization and Machine Learning, Addison-Wesley, Boston.
- Gomez, J.**, 1998. El Dessarrollo Historico de la Vivienda en Aguascalientes, Talleres de Impresion y Diseno, Mexico.
- Göloğlu, C. & Arslan, Y.**, 2006. Genetik Programlama ile İmalat için Yüzey, Pürüzlülük Modeli Geliştirilmesi, *Gazi Üniversitesi Mühendislik ve Mimarlık Fakültesi Dergisi*, **21-4**, 667-674.
- Habraken, N. J.**, 1972. Supports, an Alternative to Mass Housing, The Architectural Press, London.
- Haupt, R. & Haupt, S.E.**, 2004. Practical Genetic Algoritihms, John Willey & Sons, New Jersey.
- Hecker, D.**, 2006. Dry-In House: a Mass Customized Affordable House for New Orleans, SiGraDi2006/Medio Ambiente, Preservacion y Sustentabilidad.
- Holland, J. H.**, 1975. Adaptation in Natural and Artificial Systems, Ann Arbor: The University of Michigan Press.
- İşlier A.A.**, 2001. Üretim Hücrelerinin Bir Genetik Algoritma Kullanılarak Oluşturulması, *Anadolu Üniversitesi Bilim ve Teknoloji Dergisi*, **2-1**, 137-157.
- Kanal, L. & Cumar, V.**, 1988. Search in Artificial Intelligence, Springer, Verlag.
- Kieran, S. & Timberlake, J.**, 2004. Refabricating Architecture: How Manufacturing Methodologies are Poised to Transform Building Construction, McGraw-Hill, London.

- Kolarevic, B.**, 2003. *Architecture In The Digital Age, Design And Manufacturing*, Spon Press, New York.
- Koza, J.**, 1992. *Genetic Programming: On the Programming of Computers by means of Natural Selection*, The MIT Press, USA.
- Kurt, M. & Semetay, C.**, 2001. Genetik Algoritma ve Uygulama Alanları, *Mühendis ve Makine*, **501**, 19-24.
- Langdon, W.B. & Poli, R.**, 2002. *Foundations of Genetic Programming*, Springer Verlag, Berlin, Germany.
- Lampel, J. & Mintzberg, H.**, 1996. Customising Customization, *Sloan Management Review*, **Autumn**.
- Larson, K., Intille, S., McLeish, T., Beaudin, J., Williams, R.**, 2004. Open Source Building – Reinventing Places of Living, *BT Technology Journal*, **22-4**.
- Le Corbusier**, 1923. *Towards a New Architecture*, Dover Publications, INC, New York.
- Lohn, J. & Reggia, J.**, 1995. Discovery of Self-Replicating Structures Using a Genetic Algorithm, *1995 IEEE Int. Conf. on Evolutionary Computation (ICEC '95)*, Perth, Western Australia, **1-1**, 678-683.
- Lynn, G.**, 2000. Greg Lynn: Embryological Houses, AD – Contemporary Process in Architecture, John Wiley & Son, London, 26-35.
- Michalewicz, Z.**, 1996. *Genetic Algorithm +Data Structure = Evaluation Programs*, Third Edition, Springer Verlag, USA.
- Noguchi, M. & Hernandez Velasco, C.**, 2005. A Mass Custom Design Approach To Upgrading Conventional Housing Development In Mexico, *Habitat International* **29**, 325-336.
- Piller, F. T., & Kumar, A.**, 2006. Mass customization: Providing Custom Products and Services with Mass Production Efficiency, *Journal of Financial Transformation, Finance Factory*, **18**, 25-31.
- Pine, B. J.**, 1993. *Mass-Customization: The Frontier in Business Competititon*, Harward Business School, Boston-Massachusetts, ABD.
- Rechenberg, I.**, 1973. Evolution strategy: optimization of technical systems according to the principles of biological evolution (in German), Frommannn-Holzboog, Stuttgart.
- Rooke S.**, 1999. EvoNews:The Nerwork of Excellence in Evolutionary Computing, **11-Summer**, 9-10.
- Rosenman, M. A.**, 1996. The Generation of Form Using an Evolutionary Approach. Artificial Intelligence in 1996, (Gero, J. S. & Sudweeks, F., Eds.) Kluwer Academic, Dordrecht, Germany, 643-662.
- Smith, C.**, 1998. *Building Your Home: An Insider Guide*, Home Builder Press, Washington.
- Sims, K.**, 1994. Evolving Virtual Creatures. In Computer Graphics, *Annual Conference Series, (SIGGRAPH '94 Proceedings)*, 15-22.

Sinha N., Chakrabarti R. and Chattopadhyay, P.K., 2003. Fast evolutionary programming techniques for short term hydro-thermal scheduling, *Electric Power System Research*, **66**, 97-103.

Tipple, G., 2000. Extending themselves: User-initiated transformations of government-built housing in developing countries, Liverpool University Press, Liverpool.

Yeniay Ö., 2001. An Overview of Genetic Algorithms, *Anadolu Üniversitesi Bilim ve Teknoloji Dergisi*, **2-1**, 37-49.

Yurtçu Ş., İçağa Y., 2006. Evrimsel Algoritmaların İnşaat Mühendisliği Sistemlerinde Kullanımı, *Yapı Teknolojileri Elektronik Dergisi*, **1**, 51-59.

Yolovich, B. G., 1993. Mass-Customization Sparks Sea Change, *Business Marketing*, **November**, 43.

Url-1 <<http://www.archdaily.com/15400/acoustic-barrier-onl>>, alındığı tarih 18.01.2010

Url-2 <<http://www.goldenhook.fr>>, alındığı tarih 23.01.2010

Url-3 < <http://www.marmolradzinerprefab.com>>, alındığı tarih 24.03.2010

Url-4 <<http://www.createabook.com>>, alındığı tarih 23.01.2010

Url-5 <<http://youbars.com>>, alındığı tarih 24.01.2010

Url-6 <<http://ddbnoia.org>>, alındığı tarih 25.02.2010

Url-7 < <http://www.physicaldesignco.com>>, alındığı tarih 25.02.2010

Url-9 < <http://www.glform.com> >, alındığı tarih 24.03.2010

Url-10 <<http://www.cca.qc.ca/en/collection/6-greg-lynn-embryological-house>>, alındığı tarih 24.03.2010

Url-11 < <http://www.embryological.com>>, alındığı tarih 24.03.2010

Url-12 < <http://www.amandalevetearchitects.com/news/>>, alındığı tarih 21.04.2010

Url-13 <http://obviousmag.org/archives/2004/11/casa_citraen.html>, alındığı tarih 05.05.2010

Atalağ, K., (2001). Yaşam, Genler ve Bilgisayarlar, alındığı tarih 13.04.2010, <http://www.genbilim.com/content/view/4071/52/>

Case Study: Garofalo Architects, (2007). Collecting, Archiving and Exhibiting Digital Design Data, Art Institute Of Chicago, DAArch (Digital Archive for Architecture) Search Report, alındığı tarih 15.03.2010, <http://www.artic.edu/aic/depts/architecture/dddreport/1F.pdf>

Duarte, J. P., (1999). Malagueria, alındığı tarih 20.03.2010, <http://home.fa.utl.pt/~jduarte/malag/Malag/>

Elezkurtaj, T. & Franck, G., (1999). Algorithmic Support of Creative Architectural Design , alındığı tarih 24.04.2010, <http://www.iemar.tuwien.ac.at/publications/Umbau1.pdf>

- Hatunoğlu, B.,** (2010). Genetik Programlama, ODTÜ Bilgisayar Topluluğu Elektronik Dergisi, alındığı tarih 19.04.2010, <http://e-bergi.com/2010/Mart/Genetik-Programlama>
- Obitko,M.,** (1998). Introduction to genetic algorithms, alındığı tarih 13.04.2010, <http://www.obitko.com/tutorials/genetic-algorithms>
- Üçoluk, G.,** (2002). Evrimsel Bilgi İşleme, alındığı tarih 15.04.2010, <http://www.ceng.metu.edu.tr/~ucoluk/research/e>

ÖZGEÇMİŞ

Özge Güngör, 27 Temmuz 1984 tarihinde Antalya’da doğdu. Orta okul ve lise öğrenimini Malatya Anadolu Lisesi’nde tamamladı. 2002 yılında İstanbul Teknik Üniversitesi, Mimarlık Fakültesi, Mimarlık Bölümünde lisans eğitimine başladı. 2006 eğitim yılında Erasmus öğrenci değişim programı ile Fachhochschule Stuttgart’a gitti. 2007 senesi bahar döneminde mezun oldu. Aynı yıl İstanbul Teknik Üniversitesi, Fen Bilimleri Enstitüsü, Mimari Tasarımda Bilişim Yüksek Lisans programına kabul edildi.

İş Deneyimi:

07.2007 – Halen: Alataş Architecture & Consulting
Mimar

07/08.2006 : Demiray İnşaat
Bakırköy Park Solaire Evleri Şantiyesi – Stajyer

07/08.2005: DB Mimarlık
Ofis Stajı