TMMOB Harita ve Kadastro Mühendisleri Odası
Ulusal Coğrafi Bilgi Sistemleri Kongresi
30 Ekim –02 Kasım 2007, KTÜ, Trabzon

ÇOK-UYARLAYICILI SİSTEMDE YOL VERİLERİ İÇİN KARTOGRAFİK GENELLEŞTİRME UYGULAMASI

F. Gülgen, T. Gökgöz
Yıldız Teknik Üniversitesi, Jeodezi ve Fotogrametri Müh. Bölümü, Kartografya Anabilim Dalı, İstanbul, fgulgen; gokgoz@yildiz.edu.tr

ÖZET

Genelleştirme mevcut detaylı haritalardan veya kaynak verilerden farklı disiplinlerin ihtiyaç duyduğu daha küçük ölçekli mekansal verileri sunmayı sağlar. Kartografik genelleştirme sonucu mekansal veri çeşitli formlarda ve ortamlarda harita olarak sunulabilir. Harita ölçeği küçültüldüğünde harita işaretleri uygun olandan daha fazla yer işgal ederek mekansal çelişkilere neden olur ve sunulmak istenen verilerin okunaklılığı azalır. Bu çelişkiler yollar için kaynak veriyi basitleştirerek, biçimini değiştirerek veya abartarak giderilir. Genelleştirme çok sayıda farklı özellik arasındaki karmaşık etkileşimleri içeren olayları ele aldığından dolayı basit bir işlem değildir. Son yıllarda genelleştirme üçüncü kuşak yazılım uyarlayıcıları (agent) ile gerçekleştirilmektedir. Bunlar, kartografik görünüşlerini iyileştirmek için kendilerini genelleştiren varlıklardır. Bu çalışmada, orta ölçekli (1:25000) temel haritalardaki yol verilerinden daha küçük ölçekte (1:50000) yol verileri, çok-uyarlayıcılı (multi-agent) yaklaşım kullanılarak kartografik genelleştirmeyle elde edilmeye çalışılmıştır. Gerçekleştirilen uygulamayla son yıllarda adından oldukça sıkça söz edilen çok-uyarlayıcılı yaklaşımın gücü ve mevcut yapısındaki eksiklikler ortaya konulmaktadır.

Anahtar Sözcükler: Uyarlayıcı, Zorlayıcı, Kartografya, Harita, Ölçek

ABSTRACT

CARTOGRAPHIC GENERALISATION APPLICATION FOR ROAD DATA IN MULTI-AGENT SYSTEM

Generalisation provides to present smaller scale spatial data from the more detailed existing maps or source data for the requirements of different diciplines. The result of cartographic generalization, spatial data can be presented in several forms and environments in particular scale as map. As the scale of the map decreases in generalisation, the feature symbols cause spatial conflicts occupying a grater fraction of the available space, and the legiability of the data desired to present are decreases. These conflicts and legiability problems are resolved by simplifying, distorting or exaggerating the appearance of the source data for roads. Generalization is not a simple process because considering phenomenas which include complex interaction. In recent years, cartographic generalization is implemented with third generation software agents. They generalise themselves in order to improve their cartographic appereance. In case study, it is tried to attempt to derive smaller scale (1:50000) road data from medium scale topographic maps (1:25000) using multi-agent approach within cartographic generalisation. The implemented application shows the deficiency of multi-agent approach besides the strength.

Keywords: Agent, Constraint, Cartography, Map, Scale

1. GİRİŞ

Yeryüzü gerçekliğinin geometrik ve semantik olarak özetlenmesi anlamına gelen, farklı ölçeklerde/çözünürlüklerde gösterime olanak sağlayan genelleştirme kavramı, harita tasarımının nirengi noktalarında biri olarak düşünülmektedir. Bilgisayar teknolojilerindeki gelişmelerle birlikte genelleştirme işlemlerinin otomatik hale getirilmesi üzerine yapılan çalışmalar büyük ivme kazanmıştır. Fakat genelleştirmenin manüel tekniklerden otomatik hale getirilmesi oldukça zordur. Manüel genelleştirme, seçme, birleştirme, yumuşatma ve öteleme gibi bir dizi işlemle ve bütüncül bir yaklaşımla gerçekleştirilirken, bilgisayar ortamında işlemler oldukça karmaşık bir hal almaktadır. Bunun nedeni her bir işlem adımının tek tek ele alınması ve genelleştirme algoritmalarının birçok veriye aynı anda uygulanmasıdır (Richardson ve Mackaness, 1999).

Geometrik algoritmaların sayısal harita yazılımlarında kullanılması ve Coğrafi Bilgi Sistemi (CBS) ile birleşen otomatik genelleştirmenin ilk çalışmaları basitleştirme, öteleme, abartma, kaynaştırma ve diğer harita özelliklerini değiştirme teknikleridir. Bu yaklaşımların kısıtları, algoritmaların yalıtılmış olarak işlemesi ve işlemin uygulanabilmesi için öncelikli tetiklemelerin kartograf tarafından yapılmasıdır. Bu konuda ileri yönde atılmış en önemli adım nesne-yönelimli veri modellerinin ve mekansal araç kitlerinin tanıtılmasıdır. Bir bilgiyi kullanırken yapılan değişikliklerin komşular üzerindeki etkilerini ortaya koyan nesne-yönelimli sistemler, algoritmaların bir bağlam içinde çalışmasını sağlamıştır (1Spatial, 2007).

Günümüzde bilgisayar ortamında genelleştirme, model ve kartografik genelleştirme olarak iki farklı şekilde gruplanmıştır (Weibel ve Jones, 1998; Richardson ve Mackaness, 1999; Jiang ve Claramunt, 2004; 1Spatial, 2007).

Model genelleştirme bilgiye olan bakış açısına bağlıdır. Bu veritabanında depolanan nesnelerin sayısında yapılan bir azaltma olarak düşünülür. Veritabanında var olan veri yoğunluğunun azaltılması verinin çözünürlüğünün düşürülmesi anlamına gelir. Model genelleştirme kullanılarak temel veritabanında depolanan nesnelerin (mekansal ve semantik bilgilerin) sayısında yapılan azaltma sayesinde yeni bir veritabanı oluşturulmaktadır. Kartografik genelleştirme ise görselleştirmeye bağlıdır ve "genelleştirme sonunda elde edilen sonuçlar kartografik olarak anlaşılabilir olmalıdır" ilkesine dayanır. Kartografik genelleştirme temel ölçekten daha küçük ölçekte nesnelerin yerleştirileceği yerler ve birbirleriyle olan ilişkilerini incelemektedir. Harita tasarımı açısından bakıldığında model genelleştirme kartografik genelleştirmeden bağımsız düşünülemez. Kartografik genelleştirme istenilen mekansal verilerin okunaklı ve açık olarak görülebileceği sonuç haritanın sağlanması için bir ihtiyaçtır. Model genelleştirmeyi kartografik genelleştirme izler (Richardson ve Mackaness, 1999).

Kartografik genelleştirmenin temelinde zorlayıcılar (constraints) yatmaktadır. Zorlayıcılar, haritanın amacı, ölçek, harita işaretleri, ortam ve veri kalitesinin yanında harita üretim işlem adımlarında uyulacak mantık sırasına bağlı olarak tanımlanabilir. Semantik karmaşıklıktan dolayı tüm zorlayıcılara genel bir yöntem kullanarak cevap vermek imkansızdır. Bir zorlayıcının diğeri üzerine nasıl etki edeceği ve genelleştirme işleminin tümünün bu zorlayıcıdan nasıl etkileneceği, kontrollerin nasıl yapılacağı tam olarak tanımlanamamaktadır (AGENT Consortium, 1998).

Son yıllarda kartografyanın en karmaşık problemlerini ele alan genelleştirme konusunda çözümler üretilebilmesi için yapay zeka tekniklerinden yoğun olarak yararlanılmaktadır. Bir dağıtık yapay zeka tekniği olan çok-uyarlayıcılı (multi-agent) sistemler, çözülmesi güç problemler karşısında özerk karar verme yeteneği ile kartografik genelleştirme çalışmalarında tercih edilen bir yaklaşım olmuştur (Başaraner ve Selçuk, 2007).

Bu konuda tamamlanmış en önemli proje 1997–2000 yılarında gerçekleştirilen Avrupa Birliği destekli "Yeni Otomatik Genelleştirme Teknolojisi (Automated GEneralization New Technology – AGENT)"dir. Proje, Fransa Harita Kurumu (Institut Geographique National France - IGNF) COGIT Laboratuarı liderliğinde genelleştirme konusunda uzman Edinburgh ve Zurih Üniversiteleri ile yapay zeka konusunda uzman Grenoble Üniversitesi'nin katılımıyla gerçekleştirilmiştir. O zamanki adıyla Laser-Scan, şimdiki adıyla 1Spatial firması nesne-yönelimli LAMPS2 CBS yazılımıyla projenin endüstriyel ortaklığını yürütmüştür. Projenin sonucunda genelleştirmeye otomatik çözümler sağlayan ilk AGENT genelleştirme yazılımı örneği ortaya çıkarılmıştır (Lecordix vd., 2005). Bu projede genelleştirme üç aşamada gerçekleştirilmektedir (Lamy ve Ruas, 1999):

- *Analiz:* Harita içeriklerinin analiz edildiği ve coğrafi nesne sınıfları arasından devralınacak dokuların belirlendiği aşamadır.
- *Sentez:* Haritada genelleştirmeye ilişkin çözümlerin üretilebilmesi için nesnelerin düzenlenmesine olanak sağlayan genelleştirme aşamasıdır.
- *Değerlendirme:* Genelleştirme sonucu oluşturulan harita tasarımının başarısını ve uygunluğunu denetleyen, çözümlerin değerlendirildiği aşamadır.

Bu aşamalarda önemli olan tüm haritadan en küçük detaya kadar çeşitli kademelerde model tasarımına olanak sağlayan bir çatının oluşturulabileceği yöntemlerin belirlenmesidir (Lamy ve Ruas, 1999).

AGENT projesinden sonra, Danimarka Harita Kurumu (Danish Kort & MatrikelStyrelsen - KMS), 1Spatial firmasıyla anlaşarak, proje sonuçlarını kendi bina verileri üzerine denemiş ve 1:10000 ölçekli (TOP10DK) verilerden 1:50000 ölçekli bina verilerini elde etmeyi denemiştir. IGNF 1999 yılında AGENT projesine paralel olarak yeni bir genelleştirme projesine başlamaya karar vermiştir. "Carto2001, Space Cartographic Odyssey" isimli bu yeni proje 1:50000 ölçekli (BDCarto®) veritabanından 1:100000 ölçekli harita serilerinin üretimini amaçlamaktadır (Lecordix vd., 2005).

Bu iki endüstriyel tecrübede elde edilen başarılar her iki harita üretim kurumunun genelleştirme giderlerinin azalmasını sağlamıştır. Böylece IGNF ve KMS, 2001 yılında bu yöndeki genelleştirme çalışmalarına devam etme kararı almıştır. Belçika Harita Kurumu (Institut Geographique National Belgium - IGNB) ve İngiliz Harita Kurumu (Ordance Survey - OS) da genelleştirme çalışmalarına katılarak daha iyi bir platformda ortaklık geliştirme kararı almışlar ve 2002 yılında "Mapping Agencies Generalisation NETwork (MAGNET)" projesinin ilk adımını atmışlardır. Bu proje içinde, 2003 yılında çok-uyarlayıcılı sisteme sahip otomatik genelleştirme yazılımı olan Clarity'nin ilk sürümü üretilmiştir (Lecordix vd., 2005).

Bu çalışmada genelleştirme işlemleri 1Spatial firması tarafından üretilen Clarity programının son sürümü kullanılarak yapılmıştır.

2. ÇOK-UYARLAYICILI SİSTEM

Uyarlayıcı (agent) kavramıyla, basit olarak kendi kararlarını ve uygulamalarını kontrol eden, kendisine ve bulunduğu ortama bağlı, bir veya birden çok amaca sahip bir matematiksel varlık kastedilmektedir. Çok-uyarlayıcılı sistem ise uyarlayıcıların birbirleri ile olan etkileşimlerini düzenler (1Spatial, 2007). Bu kapsamda kartografik genelleştirme; haritalardaki nesne geometrilerinin uyarlayıcılar tarafından bir bütünlük içinde düzenlenerek, farklı ölçeklerde anlaşılır hale getirilmesi olarak da tanımlanabilir.

Uyarlayıcılar, zorlayıcılar, uyarlayıcı işlem kontrol birimi (agent process control) ve harita şartnamesi (map specifications), çok-uyarlayıcılı sistemin çekirdeğini şekillendirir. Uyarlayıcı yaşam döngüsü (agent life-cycle) ise uyarlayıcı çekirdeğinin temel faaliyetidir.

2.1 Uyarlayıcı ve Zorlayıcı Kavramları

Uyarlayıcı, bir nesnenin kartografik görünüşünü geliştirmek için kendi kendini kontrol edebilen bir varlık ise, bir uyarlayıcının eylemi, zorlayıcılar olarak bilinen ayırt edici özellikler veya kurallar dizisi tarafından tanımlanabilir. Zorlayıcılar, bir uyarlayıcının durumunu değerlendirmede ve uyarlayıcıya gerekli genelleştirme algoritmalarının uygulanmasını sağlamada kullanılır.

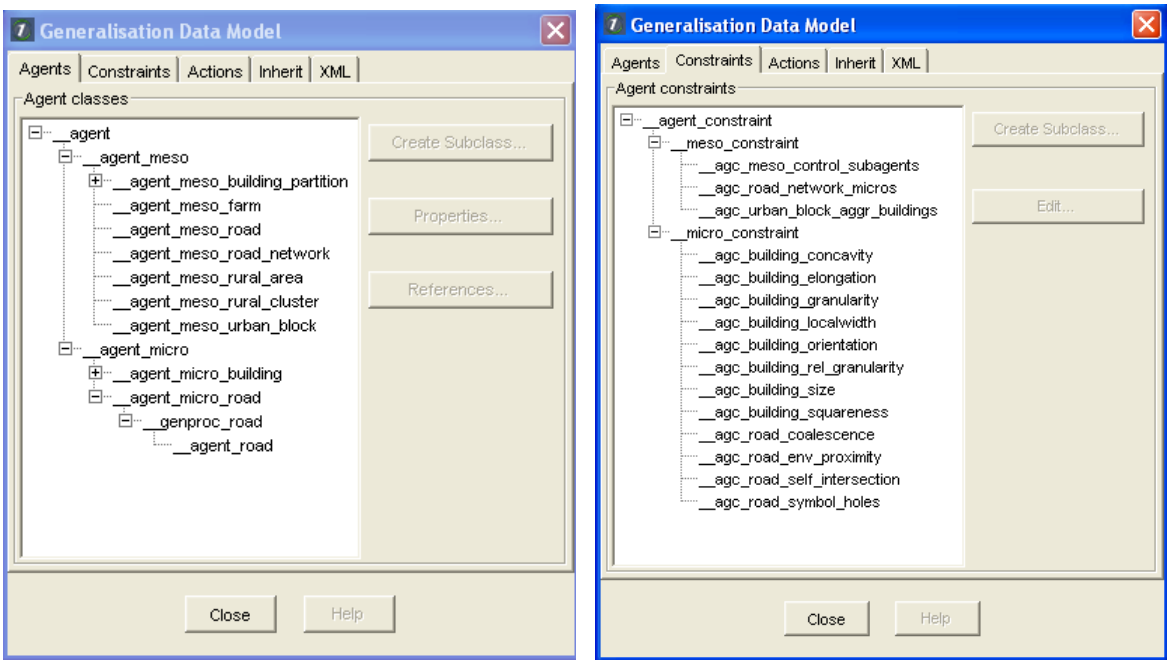
Uyarlayıcı, amacı olan bir varlık olduğundan, amaçlanan değere ulaşmak için bağımsız olarak hareket etme yeteneğine sahiptir. Bir uyarlayıcının amacı, zorlayıcılara uygunlukla orantılı olarak değiştiği varsayılan mutluluk değerini artırmaktır. Çok-uyarlayıcılı sistemde iki çeşit uyarlayıcı bulunur:

- *Yalın uyarlayıcı (micro agent):* Bir yalın uyarlayıcı, örneğin yol, bina, yazı gibi nesnelere dönüşerek kendi kendini kontrol eden ayrık bir varlıktır.
- *Bütünleşik uyarlayıcı (meso agent):* Bir bütünleşik uyarlayıcı, örneğin bir yol ağı ve bina grubundaki yalın uyarlayıcıların eylemlerini kontrol ve koordine eden geçici bir varlıktır.

Her uyarlayıcı genelleştirme davranışına kılavuzluk eden bir grup zorlayıcıya sahiptir. Her zorlayıcı sınıfı normalde özel bir uyarlayıcı sınıfının ayırt edici bir özelliğini (Örneğin; bir binanın kareselliği veya bir yazının boyu vb.) kontrol eder. Çok-uyarlayıcılı sistemde kullanılan üç çeşit zorlayıcı vardır:

- *İzleme (monitoring):* Uyarlayıcının işleyişini değerlendirir.
- *Tavsiye (proposing):* Uyarlayıcının mutluluğunu hesabeder ve eğer uyarlayıcının mevcut durumu yeterli değilse mutluluğu geliştirebilmek için planlar tavsiye eder.
- *Değerlendirme (evaluating):* Uyarlayıcının mutluluğu hesaplanırken kullanılır.

Şekil 1a, genelleştirme veri modeli içinde yeralan uyarlayıcı sınıflarının hiyerarşisini, 1b ise zorlayıcıları göstermektedir.

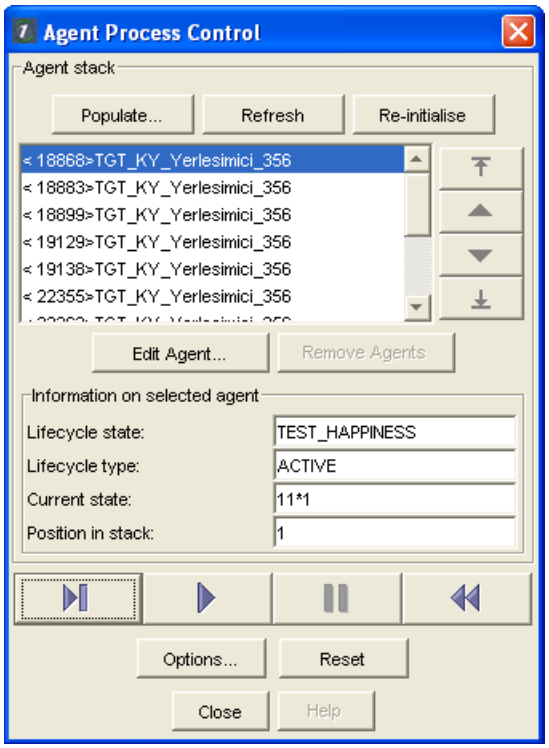


(a) (b)

Şekil 1: Uyarlayıcılar (a) ve zorlayıcılar (b)

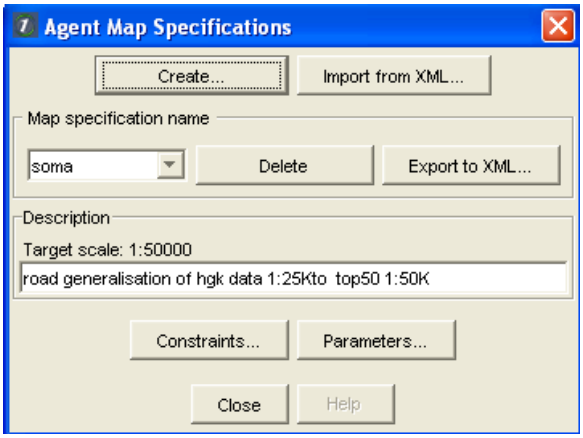
2.2 Uyarlayıcı İşlem Kontrol Birimi ve Harita Şartnamesi Kavramları

Uyarlayıcı işlem kontrol birimi, uyarlayıcıları tetikler ve tetiklenen uyarlayıcılar diğer uyarlayıcıları tetiklediğinde uyarlayıcı yığını (agent stack) güncellenir. Uyarlayıcı işlem kontrol birimi, ilgili uyarlayıcıları depolar ve yönetir. Uyarlayıcıları yığından sıra ile tetikler ve bu uyarlayıcılar yığındaki diğer uyarlayıcıların tetiklenmesine neden olur. Kullanıcı uyarlayıcı işlem kontrol birimi yardımıyla sistemin çalışmasına müdahale edebilir (Şekil 2).



Şekil 2: Uyarlayıcı işlem kontrol birimi

Harita şartnamesi, hangi zorlayıcının hangi uyarlayıcı sınıfıyla ilişkilendirildiği ve kullanılacak algoritmaların parametre bilgilerini içerir. Genellikle her hedef ölçek için bir harita şartnamesi mevcuttur. Harita şartnamesinde belli uyarlayıcı sınıflar ile ilgili zorlayıcılar; algoritmalar ve zorlayıcılar için genel ve sınıfa-özgü parametreler belirtilir (Şekil 3). Parametreler yol verilerine uygulanacak genelleştirme miktarını belirler. Parametre oranları her yol sınıfı için geçerli işaret genişliğine göre test edilerek ayrı ayrı belirlenmelidir.



Şekil 3: Harita şartnamesi

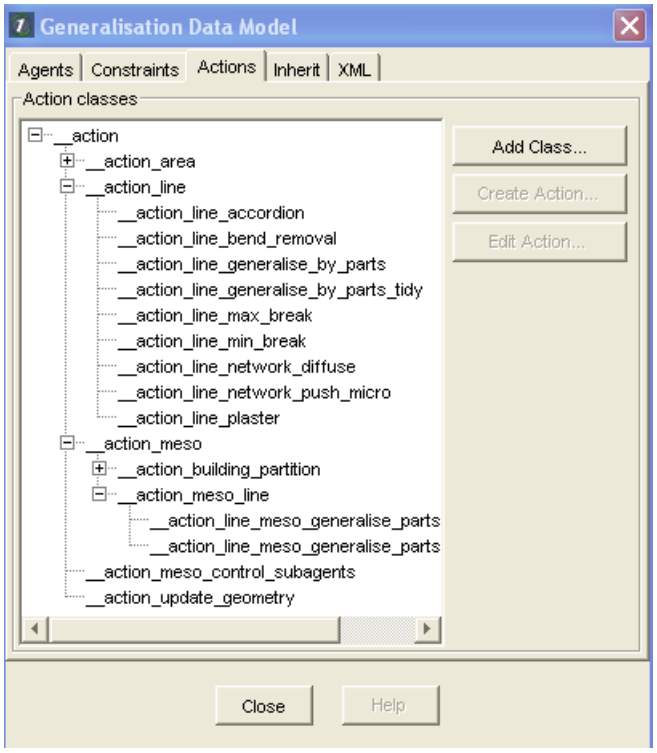
2.3 Diğer Kavramlar

Çok-uyarlayıcılı sistemin çekirdeğini oluşturan uyarlayıcı, zorlayıcı, uyarlayıcı işlem kontrol birimi ve harita şartnamesi kavramlarının dışında sistemin işleyişini açıklayabilmek için aşağıdaki kavramlardan da bahsedilmelidir.

Algoritmalar (algorithms): Clarity'de kullanılan uyarlayıcı sisteminde algoritma, bir uyarlayıcının kartografik görünümünü değiştirmek için uygulanan matematiksel işlemlerdir. Genellikle bir nesne veya nesne grubunu geometrik ve topolojik olarak düzenlemeye (Örneğin; bir çizgi geometrisini değiştirmek ya da nesnelerin yerlerini ötelemek, vb.) yarar.

Ölçüler (measurements): Nesne veya nesne grupları hakkında bilgi elde etmek için kullanılan algoritmalardır. Bir uyarlayıcının gelişmesinde ihtiyaç duyulanları ortaya çıkarmak için zorlayıcılar ölçüleri kullanır. Örneğin; bir zorlayıcı, bina büyüklüklerini hesaplayan algoritmalarla elde edilen ölçüleri en küçük büyüklük değeriyle karşılaştırır.

Eylemler (actions): Bir uyarlayıcının geometrisini değiştirmek için gerekli algoritmanın çağrılmasıdır. Şekil 4, çok-uyarlayıcılı sistemde yeralan algoritmalara ait eylemleri göstermektedir.



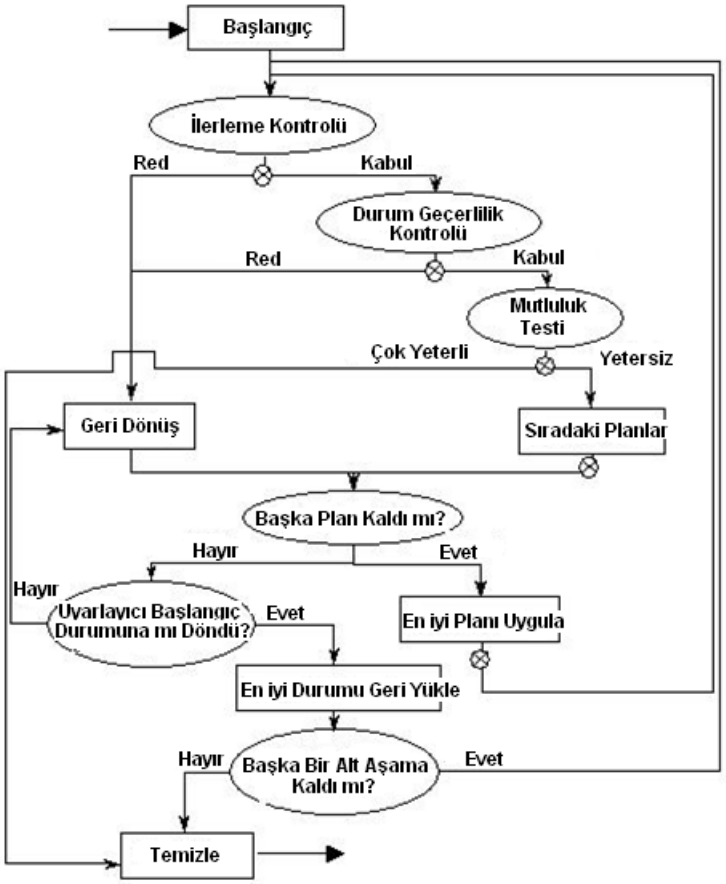
Şekil 4: Eylemler

Planlar (plans): Bir uyarlayıcıya uygulanabilecek eylemlerin ve bu eylemlerin ihtiyaç duyduğu parametrelerin tutulduğu sıralı listedir. Planlar uyarlayıcının zorlayıcılar tarafından ortaya konulan özelliklerini geliştirmek için yine zorlayıcılar tarafından önerilir.

Durum (state) ve aşama (stages): Yaşam döngüsü boyunca bir uyarlayıcı sayısız duruma girer. Uyarlayıcı yaşam döngüsü boyunca birçok ara duruma gidiyor olmasına rağmen, yaşam döngüsünün sonunda yeni geçerli yerindedir. Uyarlayıcılar farklı eylemler denediğinde bu ara aşamalar bir hiyerarşik yapı ortaya koyar. Yeni bir duruma giren uyarlayıcı her defasında bir eylem gerçekleştirmek zorundadır. Durumlar aşamalar içinde gruplandırılmıştır. Her aşama bir uyarlayıcı üzerinde yaşam döngüsünün tam bir icrasına karşılık gelir. Her bir aşama uyarlayıcının yaşam döngüsü sonlandığındaki geçerli duruma sahip olur.

2.4 Uyarlayıcı Yaşam Döngüsü

Zorlayıcıları göz önüne alarak bir uyarlayıcının görsel anlamda en iyi konuma veya şekle gelmesine neden olabilecek algoritmaların seçilerek uygulanması uyarlayıcı yaşam döngüsünde sağlanır. Şekil 5 uyarlayıcı yaşam döngüsünü göstermektedir. Bir uyarlayıcı yaşam döngüsüne girdiğinde, ilk olarak onun mevcut konumundaki memnuniyeti incelenir ve zorlayıcı ağırlıklandırmaları kullanılarak mutluluk kademelerine karar verilir. Eğer uyarlayıcı mevcut konumundan yeteri derecede mutluysa yaşam döngüsü sonlandırılır. Diğer durumda döngü işlemeye devam eder.



Şekil 5: Uyarlayıcı yaşam döngüsü (1Spatial, 2007).

Yaşam döngüsü içinde her zorlayıcı uyarlayıcı tarafından kullanılabilecek sayısız plan önerebilir. Bir uyarlayıcı için plan listesi zorlayıcılar tarafından derlenir ve planlar uyarlayıcı yığını içinde zorlayıcılar tarafından belirlenen ağırlıklandırılmış sıraya göre depolanır.

Uyarlayıcı her planı ağırlığı en yüksek olan plandan başlayarak sırayla icra eder ve yeni durumdaki sonuçları depolar. Bu yeni durumundaki uyarlayıcı zorlayıcılarını yeniden değerlendirir; önerilen plandaki zorlayıcının memnuniyetini hesaplar ve bunu, uyarlayıcının kendisi için bir bütüncül mutluluk seviyesini hesaplamada zorlayıcının önemi ile birlikte kullanır. Elde edilen değerler, yeni durum uyarlayıcıya beklenen derecede mutluluk sağlamasa bile, önceki yere göre muhtemel bir gelişme olup olmadığı açısından kontrol edilir. Bu kontrol sonucu eğer uyarlayıcının durumunda bir gelişme yoksa uyarlayıcı planın devreye girmesinden önceki yerine geri döner ve bir sonraki gelecek planı dener. Eğer bir gelişme varsa uyarlayıcı şimdiki yerinde önerilen yeni planları denemeye devam eder.

Belli bir durum için önerilebilecek plan kalmadıysa uyarlayıcı bir önceki yerine geri döner. Bu durumda eğer mevcut durum için henüz icra edilmeyen yeni planlar varsa yaşam döngüsü işlemeye devam eder.

Yaşam döngüsünün sonunda uyarlayıcı ilk durumuna geri döner ve mutluluk derecesinin en yüksek olduğu yer uyarlayıcının yeni durumu olarak tekrar yüklenir.

3. ÇOK-UYARLAYICILI SİSTEMDE YOL VE YOL AĞI GENELLEŞTİRMESİ

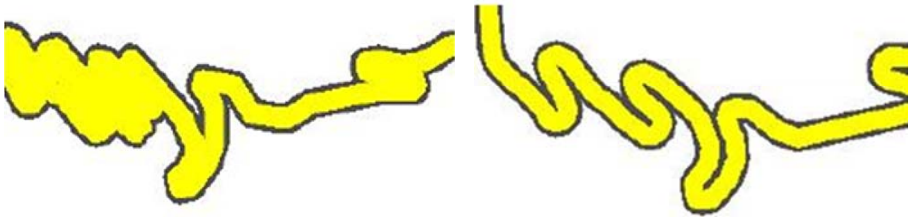
Haritadaki yol genişliklerinin büyük oranda abartılarak gösterilmesi kartografik genelleştirmeye anahtar bakış açısıdır. Örneğin; 5 m. genişliğindeki bir şehiriçi yolun haritada 1 mm. ile gösterilmesi bunun yeryüzünde on kat büyüyerek 50 m.'lik yol genişliğini ifade etmesi anlamına gelebilmektedir. Bu durumda okunabilirlik kartografik genelleştirme ile sağlanır.

Çok-uyarlayıcılı sistemde yol verileri için bağımsız ve çevresel genelleştirme söz konusudur. Bağımsız genelleştirmede tek bir yol nesnesine ait okunabilirlik sorunları giderilirken, çevresel genelleştirmede yol ağını oluşturan nesneler arasında meydana gelen sorunlar (Örneğin; aşırı yol yoğunluğu, yolların örtüşmesi, vb.) çözülmektedir (Duchene vd., 2001).

Uyarlayıcılarda kullanılan iki temel seviye olan yalın ve bütünleşik seviyeler yol verileri için de geçerlidir. Veritabanı düşünüldüğünde, bir yol parçası tek başına yalın seviyeyi oluşturur. Yol ağı ise bütünleşik seviyededir. Buna karşın yalın seviyedeki bir yol, genelleştirme sırasında yalın seviyede yeni yol parçalarına ayrılmak zorunda kalabilir. Böylece alt yalın seviyeler oluşur. Bütünleşik seviyedeki nesneler veri tabanında doğrudan saklanmasa da mekansal analizlerle ortaya çıkarılabilir (Duchene vd., 2001).

Zorlayıcılar da aynı şekilde iki farklı seviyede ele alınmaktadır. Yalın yol seviyesinde farklı ölçeklerde okunabilirliğe ilişkin iki temel işaretleme sorunu (birleşme ve aşırı detay) zorlayıcı olarak düşünülmektedir (Duchene vd., 2001).

- *Birleşme (coalescence)*: Bir yol işaretinin kendi üzerine kapanması (Şekil 6),
- *Aşırı detay (granulity):* Görüntülenecek ölçekte bir çizginin aşırı sayıda detay noktasından oluşması.



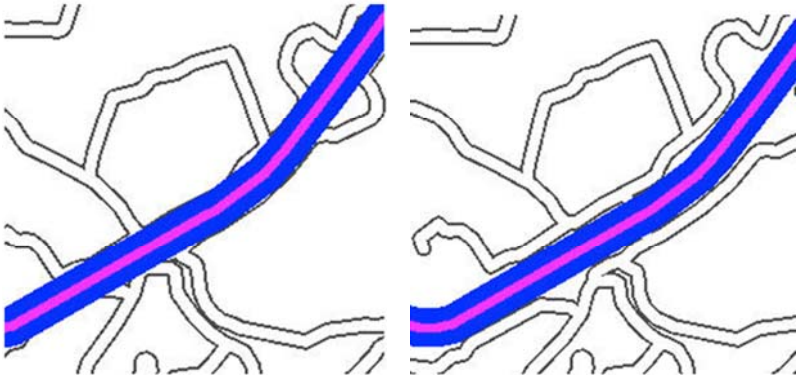
(a) (b)

Şekil 6: Genelleştirme öncesi (a) ve sonrası (b) çok kıvrımlı yol (1Spatial, 2007).

Bu seviyede okunabilirlik sorunlarını gidermek için, genelleştirmede şekil ve konum doğruluğu önceliklidir ve uygun değişikliklerin kullanıcı tarafından tanımlanması beklenir. Genelleştirme sonucu uyumsuzluklar giderilmelidir.

Bütünleşik yol ağı seviyesinde yol işaretleri, yakın yol parçaları arasında üst üste binerek örtüşmelere neden olur (Şekil 7a). Bu durumda küçük yol parçaları gizlendiğinden (altta kaldığından) gerçekte olmayan yeni kavşak yapıları meydana gelir. Bu problem yanlış topolojik ilişkilerin kurulmasına neden olur.

Yol ağı seviyesinde oluşan sorunları gidermek için, yollar ötelenerek minimum okunabilir mesafeler sağlanır (Şekil 7b). Böylece topolojik karakteristiklikler korunabilir. En kötü şartlarda bozulmalara neden olan önemsiz yollar ortadan kaldırmalıdır.



(a) (b)

Şekil 7: Yol ağı kavşaklarının genelleştirme öncesi (a) ve sonrası (b) durumu (1Spatial, 2007).

3.1 Yol Genelleştirmesinde Kullanılan Algoritmalar

AGENT projesinde toplam sekiz yol genelleştirme algoritması kullanılmıştır. Bunlardan altısı dönüşümlerle ilgiliyken, biri yol çizgilerini parçalara ayırmada, diğeriyse genelleştirme etkilerinin yayılmasında (bütünü oluşturan diğer parçalara yansıtılmasında) kullanılmıştır.

- *Dönüşüm algoritmaları:* Tek yönde kıvrımlı yollar için kullanılan *en büyük kıvrım (maximal break)* ve *en küçük kıvrım (minimal break)* algoritmaları (Şekil 8); seri kıvrımlı yollar için kullanılan *akordeon (accordion)* ve *kıvrım azaltma (bend removal)* algoritmaları (Şekil 9); tüm yollara uygulanabilen *Gaussian yumuşatma (Gaussian smoothing)* ve *sıvama (plaster)* algoritmaları (Şekil 10).
- *Parçalama (focalisation) algoritması* (Şekil 11).

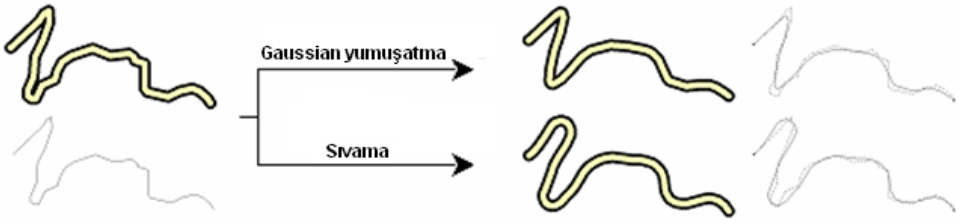
- *Yayma (propagation) algoritması* (Şekil 12).



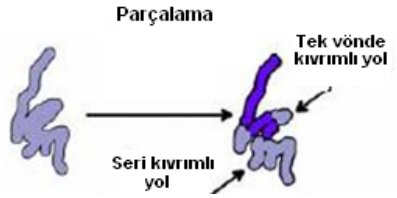
Şekil 8: Tek yönde kıvrımlı yollar için kullanılan dönüşüm algoritmaları (Duchene vd., 2001).



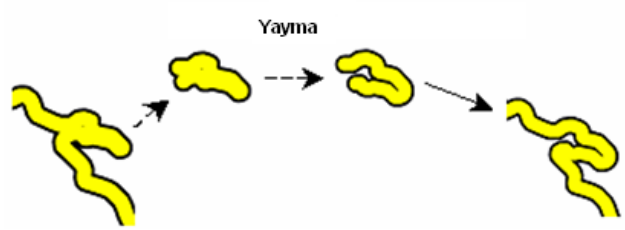
Şekil 9: Seri kıvrımlı yollar için kullanılan dönüşüm algoritmaları (Duchene vd., 2001).



Şekil 10: Yumuşatma algoritmaları (Duchene vd., 2001).



Şekil 11: Parçalama algoritması (Duchene vd., 2001).



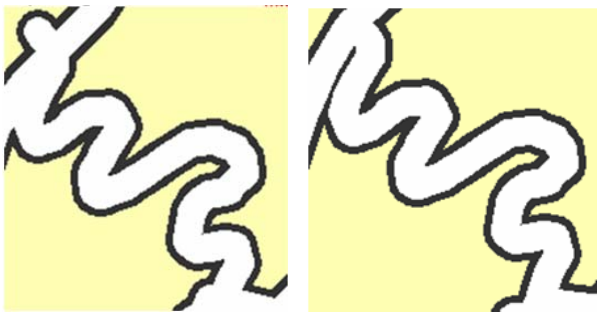
Şekil 12: Yayma algoritması (Duchene vd., 2001).

3.2 Yol Ağı Genelleştirmesinde Kullanılan Algoritmalar

Yol ağlarının genelleştirilmesinde birçok algoritma kullanılmaktadır. Bu algoritmalardan ikisi aşağıda verilmiştir.

Elastik çubuklar (beams) algoritması: Bu algoritma yol ağında yol işaretlerindeki örtüşmelerden kaynaklanan yeni kavşak oluşumlarını engeller. Yollar ağdaki açıklığın sağlanabilmesi için üst üste binmelerden ötelenerek kurtarılır. Şekil 7 yol ağlarında kavşak noktalarının elestik çubuklar algoritması uygulanmadan önceki ve sonraki durumu göstermektedir.

Dayama (diffusion) algoritması: Bu algoritma genelleştirme süresince yol ağının mevcut bağlantılarını korumaya yarar. Bir yola yalın seviyedeki genelleştirme algoritmaları uygulandıktan sonra, bu yola bağlantılı yolların kavşak noktalarında bozulmalar meydana gelebilir. Bu durumda dayama algoritması kavşak yapılarının korunmasını sağlar (Şekil 13).

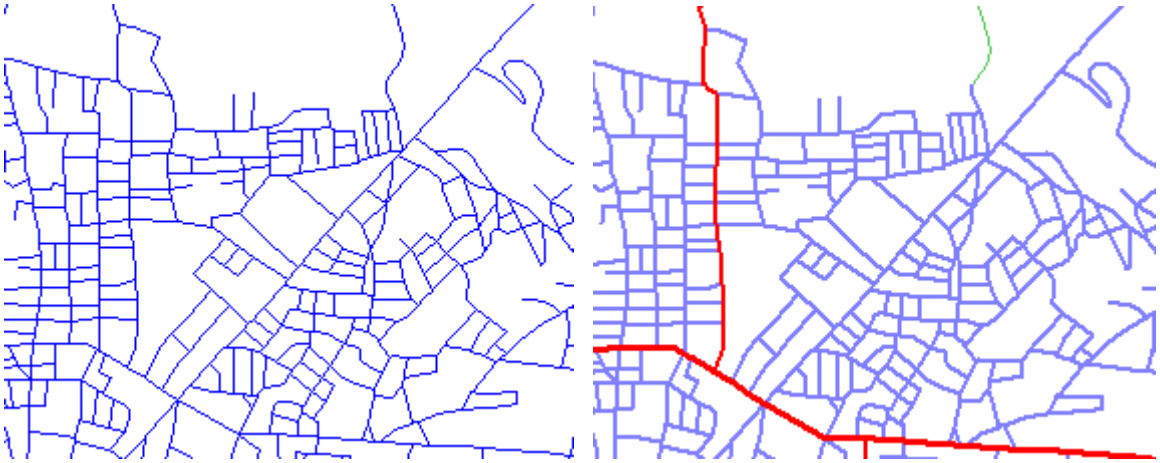


(a) (b)

Şekil 13: Dayama algoritması uygulanmadan önceki (a) ve sonraki (b) durum (IGNF verilerinden).

4. UYGULAMA

Şekil 14a'da çalışmada kullanılacak, temel veritabanında yeralan 1:25000 ölçekli yol verilerinin küçük bir kısmı ölçekli olarak görülmektedir. Yol verileri Clarity genelleştirme programının kullandığı nesne yönelimli GOTHIC veritabanı içinde karayolu ve demiryolu olarak iki ana sınıf altında depolanmıştır. Şekil 14a'nın içerdiği bölgede sadece karayolu verileri yeralmaktadır. Bu veriler, Harita Genel Komutanlığı'nın (HGK) 1:25000 ölçekli haritalarda kullandığı yol işaret genişliklerine göre altı alt sınıfa ayrılmıştır. 1:25000 ölçekli topografik haritalara uygun yol işareti giydirilmiş veriler haritada Şekil 14b'de olduğu gibi görülmektedir.



(a) (b)

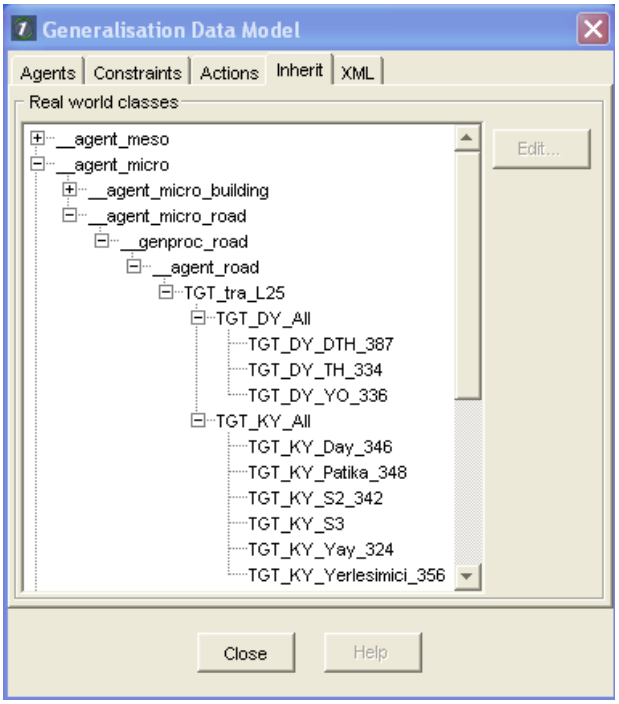
Şekil 14: 1:25000 ölçekli yol verilerinin veritabanındaki (a) ve haritadaki (b) görüntü.

Yol işaret genişlikleri 1:25000 ölçeğinden 1:50000 ölçeğine dönüşürken iki kat büyümektedir. Şekil 15 aynı bölgenin 1:25000'lik altlıkta 1:50000 ölçekli yol işaretleri giydirilmiş durumunu göstermektedir. Bu durumda yol işaret genişlikleri büyüdüğünden, işaretlerin 1:25000 ölçekli altlıkta kaplayacağı alan artmakta ve haritada işaretlerin üst üste binmesinden kaynaklanan okunabilirlik sorunları meydana gelmektedir.



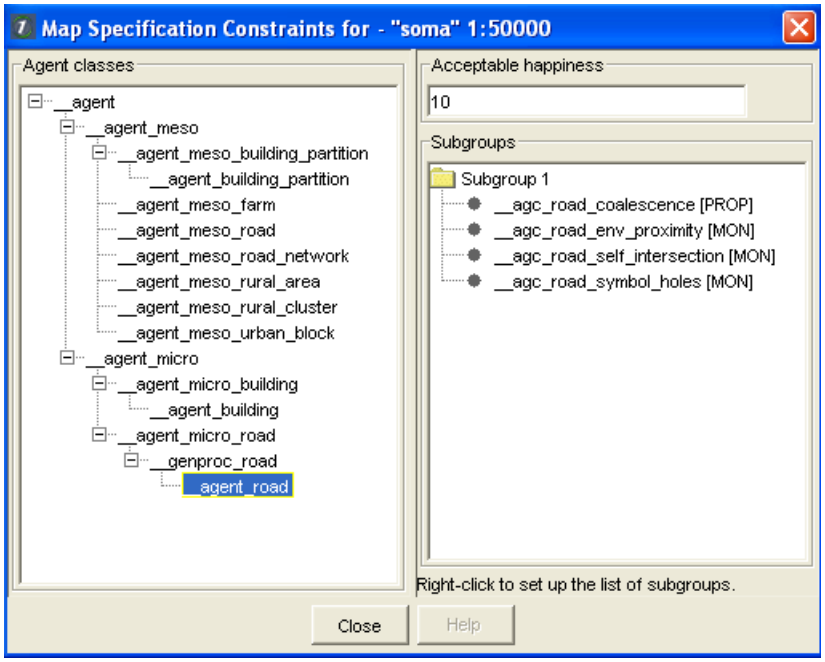
Şekil 15: 1:25000 ölçekli altlıkta 1:50000 ölçekli yol işareti giydirilmiş görüntü.

Çok-uyarlayıcılı sistemi kullanan Clarity kartografik genelleştirme programı gerçek dünya sınıflarını oluşturan yol verilerini nesne yönelimli veritabanı içinde, kendi uyarlayıcı sınıflarının alt sınıfları olarak depolamaktadır. Şekil 1a'da görülen Clarity uyarlayıcı sınıflarının alt sınıfları olarak depolanan yol verilerinin veritabanındaki hiyerarşisi Şekil 16'da gösterilmektedir.



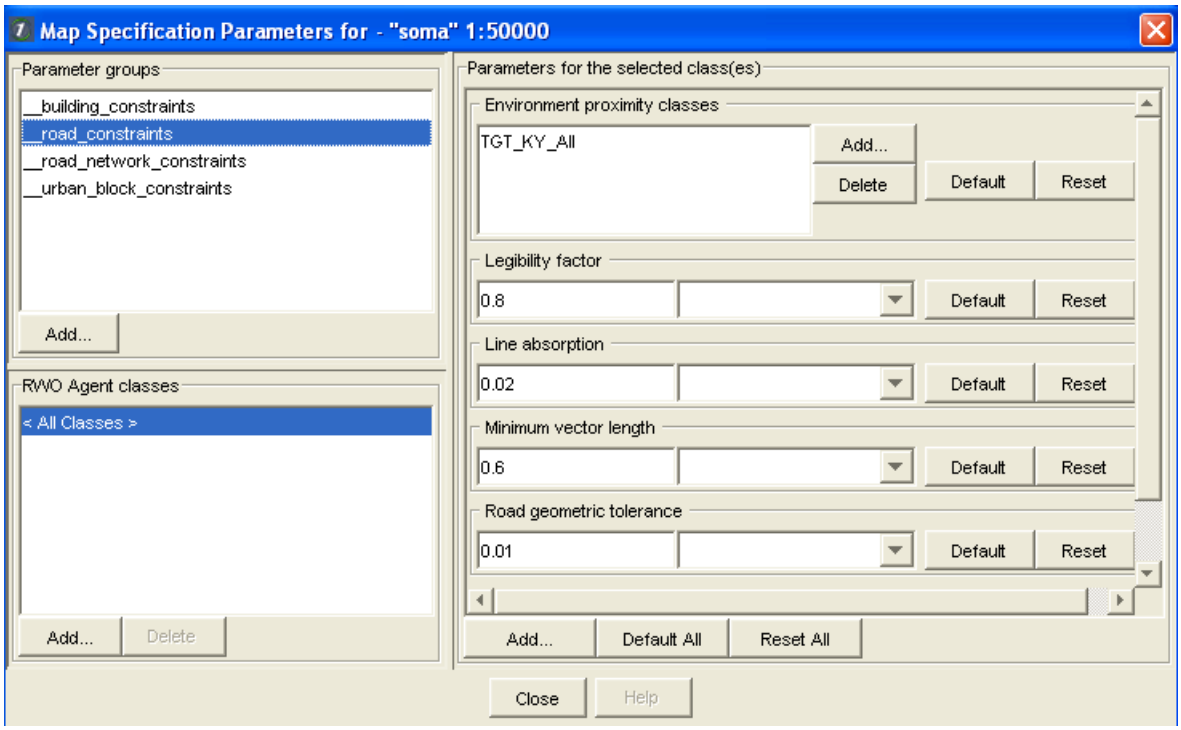
Şekil 16: Uygulamada kullanılan gerçek dünya sınıflarının hiyerarşisi.

Uygulamada kullanılan izleme ve tavsiye zorlayıcıları Şekil 17'de *Subgroup 1* altında görülmektedir.



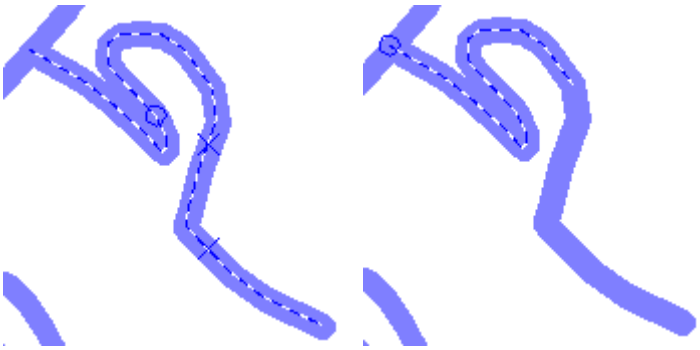
Şekil 17: Uygulamada kullanılan yol zorlayıcıları.

Şekil 18'de uygulamada kullanılan bazı parametre değerleri gösterilmiştir.



Şekil 18: Uygulamada kullanılan bazı yol parametreleri.

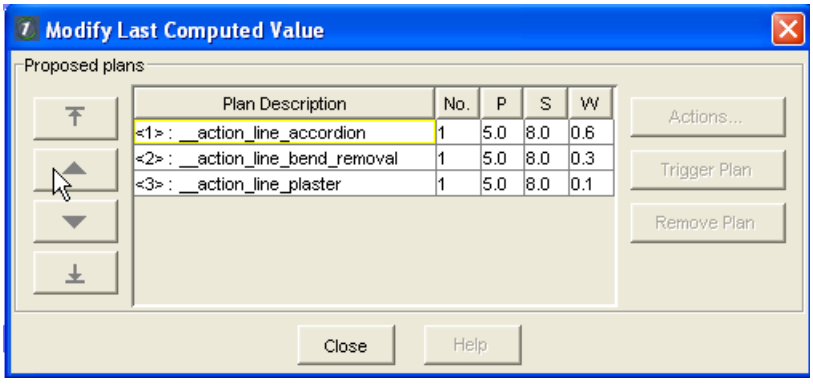
Şekil 19'da birleşme sorunu olan bir yol nesnesi görülmektedir. Genelleştirme işlemi uygulandığında yol nesnesi bir yalın uyarlayıcı tarafından parçalama algoritması ile alt yalın uyarlayıcılara ayrılmıştır. Bu parçalardan birleşmenin olduğu sorunlu kısım Şekil 19b'de görülmektedir.



(a) (b)

Şekil 19: Genelleştirme öncesi (a) ve parçalama algoritması uygulandıktan sonra (b) yol nesnesinin durumu.

Çok-uyarlayıcılı sistem tarafından önerilen, bu yol parçasına uygulanabilecek eylemler sıra ile akordeon, kıvrım azaltma ve sıvama algortmalardır (Şekil 20).



Şekil 20: Yol nesnesine uygulanması için önerilen planlar.

Bunlardan sıvama algoritması en iyi sonucu verdiğinden yol nesnesine çok-uyarlayıcılı sistem tarafından uygulanmıştır (Şekil 21a). Daha sonra yol nesnesi tekrar birleşerek genelleştirilmiş duruma gelmiştir (Şekil 21b).



(a) (b)

Şekil 21: Sıvama algoritması (a) ve birleşme (b) sonucu yol nesnesinin durumu

Şekil 15'de 1:25000 ölçeğinde genelleştirme öncesi durumu gösterilen, 1:50000 ölçekli yol işaretleri giydirilmiş verilerinin çok-uyarlayıcılı sisteme göre uygulanan kartografik genelleştirme sonrası durumu, Şekil 22a'da görülmektedir. Görüntü iki kat küçültüldüğünde Şekil 22b'deki 1:50000 ölçekli HGK yol genişlik standartlarına uygun harita görüntüsü meydana gelmektedir.



(a) (b)

Şekil 22: 1:50000 ölçekli yol işaretlerine sahip 1:25000 (a) ve 1:50000 (b) ölçekli genelleştirilmiş görüntü.

5. SONUÇ

Yapılan uygulama çok-uyarlayıcılı sistemin bağımsız yol verileri üzerinde oldukça etkili olduğunu göstermektedir. Uygun parametreler seçilebildiği takdirde kartografik anlamda okunabilirlik sorunları büyük ölçüde ortadan kalkmaktadır. Diğer taraftan, komşu yol parçaları arasında meydana gelen üst üste binmelerden kaynaklanan sorunlar bu çalışma kapsamında uygulama ile değerlendirilememiştir. Bunun nedeni, bu sorunları ortadan kaldırdığı söylenen elastik çubuklar algoritmasının Clarity'nin çok-uyarlayıcılı sistemi içinde henüz yeralmamasıdır.

Çok-uyarlayıcılı sistemin önemli eksikliği genelleştirmenin yani ölçek değişiminin fazla olduğu durumda ortaya çıkmaktadır. Yol işaret genişlikleri 1:25000 ölçeğinden 1:100000 veya 1:250000 ölçeğine uygun işaret genişliklerine dönüştürüldüğünde, çok-uyarlayıcılı sistemde kullanılan kartografik genelleştirme algoritmalarının yeterli olacağı söylenemez. Bu tarz bir genelleştirmede harita işaretlerinde yapılacak dönüşümlerin yanında, bazı yol sınıflarının haritadan kaldırılması veya yol sınıflarından önemsiz sayılabilecek bazı yol nesnelerinin silinmesi gerekmektedir. Kartografik anlamda okunabilirlik sorunları ölçek değişiminin fazla olduğu durumlarda ancak bu yolla giderilebilir.

KAYNAKLAR

1Spatial, 2007. *Concepts in Clarity and ClearText*, Issue 1.3 for Clarity vol.2–6 Release Notes.

AGENT Consortium, 1998. *Generalisation Modelling Using and Agent Paradigm*, Project Report D A1, ESPRIT/LTR/24–939.

Başaraner M. ve Selçuk M., 2007. *Çok Ajanlı Sistemler ve Kartografik Genelleştirme*, TMMOB Harita ve Kadastro Mühendisleri Odası, 11. Türkiye Harita Bilimsel ve Teknik Kurultayı, 2-6 Nisan, Ankara.

Duchene C., Barrault M. ve Haire K., 2001. *Road Network Generalization: A Multi-Agent System Approach*, Proc. of the 20th International Cartographic Conference, vol.3, Beijing, 2166–2177.

Jiang B. ve Claramunt C., 2004, *A Structural Approach to the Model Generalization of an Urban Street Network*, GeoInformatica, Vol. 8 Issue 2, 157-172.

Lamy S., Ruas A., Demazeau Y., Jackson M., Mackaness W.A. ve Weibel R., 1999. *The Application of Agents in Automated Map Generalisation*, Proc. of 19th International Cartographic Conference, Ottawa, vol.2, 1225-1234.

Lecordix F., Regnauld N., Meyer M. ve Fecrih A., 2005. *MAGNET Consortium*, 8th ICA WORKSHOP on Generalisation and Multiple Representation, 7-8th July, A Coruna, Spain.

Richardson D.E. ve Mackaness W.A., 1999. *Computational Processes for Map Generalization*, Cartography and GIS, Vol:26-1.

Weibel R. ve Jones C. B., 1998. *Computational Perspectives on Map Generalization*, GeoInformatica, Vol. 2 Issue 4, 307–315.