TOBB EKONOMİ VE TEKNOLOJİ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

KENTSEL ALAN HAREKETLİLİĞİNDE ANONİM KONUM PAYLASIMI

YÜKSEK LİSANS TEZİ

Ozan Berk BİTİRGEN

Bilgisayar Mühendisliği Anabilim Dalı

Tez Danışmanı: Prof. Dr. Osman ABUL

NİSAN 2022

TEZ BİLDİRİMİ

Tez içindeki bütün bilgilerin etik davranış ve akademik kurallar çerçevesinde elde edilerek sunulduğunu, alıntı yapılan kaynaklara eksiksiz atıf yapıldığını, referansların tam olarak belirtildiğini ve ayrıca bu tezin TOBB ETÜ Fen Bilimleri Enstitüsü tez yazım kurallarına uygun olarak hazırlandığını bildiririm.

Ozan Berk BİTİRGEN

ÖZET

Yüksek Lisans Tezi

KENTSEL ALAN HAREKETLİLİĞİNDE ANONİM KONUM PAYLASIMI

Ozan Berk BİTİRGEN

TOBB Ekonomi ve Teknoloji Üniversitesi
Fen Bilimleri Enstitüsü
Bilgisayar Mühendisliği Anabilim Dalı

Tez Danışmanı: Prof. Dr. Osman ABUL

Tarih: NİSAN 2022

GPS donanımlı mobil cihazlar günlük hayatın ayrılmaz bir parçası olmaya başladı. Bunun sonucu olarak kullanıcılarına konum tabanlı servis (KTS) sağlayan sosyal ağların sayısı gün geçtikçe artmaktadır. KTS'ler tarafından sunulan ortak bir servis, konum güncelleme servisidir. Konum güncellemeleri coğrafik düzlemdeki gerçek konumu göstermesi sebebiyle, konum tabanlı servis üzerinden yapılan konum güncellemeleri mahremiyet sorunlarını da beraberinde getirmiştir. Bu tezde, insan hareketliliği uygulamaları için kentsel alan hareketliliğinde konum mahremiyeti korumalı konum güncellemesi problemi çalışılmış ve çözümü için veri-merkezli bir çatı önerilmiştir. Tezin ilk bölümünde, kullanıcıların konum mahremiyet profillerine göre şehir ağı üzerinde efektif bir perdelenmiş bölge hesaplama yöntemi geliştirilmiş ve bunu uygulayan bir hareketlilik modeli önerilmiştir. Sonrasında hız tabanlı saldırılar dikkate alınarak ilgili konum mahremiyet modelleri önerilmiş ve konum mahremiyet ihlali yaratan konum güncelleme isteklerini engelleyen algoritmalar geliştirilmiştir. Tezin ikinci bölümünde ise, birlikte bulunma olasılığı olan kullanıcı gruplarına yönelik ortak konumlandırma saldırıları detaylandırılmış ve konum bilgilerinin oluşturabileceği çıkarım kanalları dikkate alınarak ek bir mahremiyet modeli önerilmiş, bu modele göre konum mahremiyetini koruyan

algoritmalar geliştirilmiştir. Tez kapsamında konum k-anonimliği, konum mahremiyetini korumak için kullanılmış ve paylaşılan konumun konum mahremiyetini koruması için en az k adet düğüm içermesi olarak düşünülmüştür. Ayrıca, tez çalışması kapsamında simüle verisetleri ve yörüngeler üretilmiş, önerilen hareketlilik modelini ve algoritmaları gerçekleştiren kampsamlı deneysel çalışmalar yapılmış ve sonuçları da sunulmuştur.

Anahtar Kelimeler: Konum tabanlı servisler, Konum paylaşımı, Konum mahremiyeti, Ortak konumlandırma saldırıları, Çıkarım kanalı, Hız tabanlı saldırılar .

ABSTRACT

Master of Science

ANONYMOUS LOCATION SHARING IN URBAN AREA MOBILITY

Ozan Berk BİTİRGEN

TOBB University of Economics and Technology
Institute of Natural and Applied Sciences
Department of Computer Engineering

Supervisor: Prof. Dr. Osman ABUL

Date: APRIL 2022

GPS-equipped mobile devices are increasingly becoming an integral part of our daily lives. Consequently, the number of social networks that provide location based services (LBS) are increasing more and more. A typical service provided by these LBS's is the location check-in services. Location check-in via location-based service have brought privacy issues as each location check-in show the actual location in the geographic plane. This thesis studies the problem of privacy-preserving location update in the context of urban area people mobility. As a result, a data-centric framework is introduced. In the first section, an effective cloaking method has been developed to construct cloaking regions with respect to users' location privacy profile specification under road network. And a mobility model which utilizes these cloaking regions is proposed. Afterwards, related location privacy models have been proposed and algorithms providing location k-anonymity has been developed under road network and velocity-based attack constraints. In the second section, co-localization attacks has been detailed for co-location likely user groups and algorithms providing location k-anonymity has been developed under inference channels arising from user interactions. Within the scope of the thesis, location k-anonymity model is followed as a means to protect location privacy, i.e., the shared

location (corresponding to a subgraph) is considered to be location privacy preserving if it contains at least k vertices. In addition, simulated datasets and trajectories are generated, comprehensive experimental studies that implements the proposed mobility model and algorithms has been carried out and their results are presented as well.

Keywords: Location based services, Location sharing, Location privacy, Co-localization attacks, Inference channel, Velocity-based attacks.

TEŞEKKÜR

Makale ve tez çalışmalarım boyunca değerli yardımını esirgemeyen ve katkılarıyla beni yönlendiren değerli hocam Prof.Dr. Osman ABUL, kıymetli tecrübelerinden faydalandığım TOBB Ekonomi ve Teknoloji Üniversitesi Bilgisayar Mühendisliği Bölümü öğretim üyelerine, tez çalışmamın temelini oluşturmamı sağlayan 118E712 nolu proje kapsamında destek veren TÜBİTAK'a ve destekleriyle her zaman yanımda olan aileme ve arkadaşlarıma çok teşekkür ederim.

İÇİNDEKİLER

ŞEKİL LİSTESİ

ÇİZELGE LİSTESİ

KISALTMALAR

Casper : Konum Servisleri İçin Mahremiyetten Ödün Vermeden Sorgu İşleme - Query Processing for Location Services without Compromising Privacy
GPS : Küresel Konumlandırma Sistemi - Global Positioning System
KTS : Konum Tabanlı Servis - Location Based Service
PROBE :Mahremiyet Korumalı Şaşırtma Ortamı - Privacy Preserving Obfuscation Environment
TTP : Güvenilir Üçüncü Taraf - Trusted Third Party

SEMBOL LİSTESİ

Bu çalışmada kullanılmış olan simgeler açıklamaları ile birlikte aşağıda sunulmuştur.

Simgeler	**Açıklama**
ACN	Açıklamalı şehir ağı
ACCN	Açıklamalı kaba şehir ağı
B	Kanı fonksiyonu
CG	Ortak konumda bulunma olasılığı olan grup
CT	Kaba kullanıcı yörüngesi
CLP	Ortak konumda bulunma fonksiyonu
EPB	Ekstrapole edilmiş kanı fonksiyonu
IC	Geçerli çıkarım kanalı
k	Anonimlik seviyesi
kPACN	k-üyeli bölütlenmiş açıklamalı şehir ağı
PB	Sonraki kanı fonksiyonu
TT	Gerçek kullanıcı yörüngesi
t	Zaman bilgisi
τ	Şu anki zaman
Δt	Çıkarım kanalı geçerlilik süresi

1. GİRİŞ

Günümüz akıllı mobil cihaz teknolojisindeki gelişmeler ve bu cihazların maliyetlerinin düşmesiyle beraber kullanım miktarları oldukça artmış durumdadır. Dünya genelindeki akıllı mobil cihaz kullanıcı sayısı 2020 itibariyle 3 milyarı aşmış durumdadır ve bu rakamın önümüzdeki birkaç yıl içinde birkaç yüz milyon daha artıp 4 milyarı aşacağı tahmin edilmektedir [43]. Küresel Konumlama Sistemi (GPS) teknolojisinin gelişmesi ve internetin ulaşılabilirliğinin kolaylaşmasıyla beraber kişilerin uzay-zamansal bilgilerini kullanarak bilgi ve hizmet sağlayan Konum Tabanlı Servisler (KTS) ortaya çıkmıştır [48]. Konum bilgisi birçok farklı bilgiye erişme amacıyla kullanılabileceği için, kullanıcılarına konum tabanlı servis sunan uygulamaların sayısı ve bu servislerin kullanımı her geçen gün artmaktadır.

Konum tabanlı servisler, kullanıcıların ihtiyaçlarına göre birçok farklı alanda hizmet vermektedirler [44]. Bunlardan başlıcaları acil durum uygulamaları, navigasyon uygulamaları, bilgi alma uygulamaları, reklam uygulamaları, izleme ve yönetim uygulamaları ve sosyal ağ uygulamalarıdır. Uber, Twitter, İnstagram, Facebook, Swarm, Tinder, Google Haritalar gibi birçok uygulama kullanıcılarına konum tabanlı servisler sunmaktadır. Aracı olmayan birisi, Uber'i kullanarak kendi konumundan gitmek istediği konum için bir araç ve sürücü bulabilir. Arabayla hiç bilmediği bir yere gitmeye çalışan birisi, Google Haritalar'dan faydanalanarak gitmek istediği yere gidebileceği en kısa yolu bulabilir. Sosyalleşmek isteyen birisi, Facebook Yakındaki Arkadaşlar servisini kullanarak yakınında hangi arkadaşlarının olduğunu öğrenebilir ve istediğiyle görüşebilir.

Konum tabanlı servis sunan uygulamalar üzerinde konum bilgisinin kullanılma şekli de farklılık göstermektedir. Twitter ve İnstagram gibi sosyal uygulamalarda konum paylaşımı genellikle paylaşılan bir fotoğrafa konum etiketi koyularak yapılır. Tinder, Swarm ve Facebook gibi sosyal uygulamalarda ise kullanıcılar arkadaşlarını bilgilendirmek amacıyla anlık olarak gerçek konumlarını paylaşırlar. Google Haritalar gibi navigasyon uygulamaları ise hizmet sağlayabilmek için kullanıcıların gerçek zamanlı konum bilgisini kullanır. Ayrıca konum tabanlı servis sunmasa bile birçok uygulamanın pazarlama ve reklam amacıyla kullanıcılarının gerçek konum bilgilerini kullandığı, sakladığı ve üçüncü taraflara sattığı bilinmektedir [46]. Bu sebeple, konum

tabanlı servisler insanlara ne kadar fayda sağlasa da birçok insan bu servisleri kullanırken endişe duymaktadır.

Konum bilgisi, parmak izi veya kimlik numarası gibi kişiye özel bir özniteliktir ve bu bilgi kullanılarak kişiyle ilgili birçok çıkarımda bulunulabilir [22]. Bu sebeple konum bilgisinin paylaşılması, kişilerin fiziksel güvenliğini riske atmak ve itibarını zedelemek gibi birçok güvenlik ve mahremiyet riskini beraberinde getirir. Birçok ülkede kişilerin anlık ve tarihsel konum bilgisinin mahrem olduğu yasalarla kabul edilmiştir. Örneğin ABD'de 2012 tarihli "Location Privacy Protection Act" kanunu, konum bilgisi paylaşımını ve iletimini düzenlemektedir.

Çoğu durumda KTS'lerin sunduğu hizmetler, günlük hayatımızı oldukça kolaylaştırmaktadır ve bu hizmetlerden faydalanmak isteyen kullanıcılar, bu servislere güvenerek konumlarını gönüllü bir şekilde paylaşırlar. Fakat KTS'lerin güvenilir olmama durumu da oldukça olasıdır. Ayrıca KTS'ler güvenilir olsa bile, artık dışarıdan gelen saldırılarla kullanıcı verilerinin sızdırılması durumu çok sık görülmektedir [1]. Örneğin Haziran 2021 tarihli bir siber saldırıda, 700 milyon Linkedin kullanıcısının konum bilgisi de dahil olmak üzere birçok bilgisi sızdırılmış ve verilerin bir kısmı Dark Web üzerindeki bir forumda satışa sunulmuştur [2]. Ocak 2021 tarihli bir başka siber saldırıda ise, hızlı bir şekilde büyümekte olan Çin merkezli sosyal medya şirketi Sociallarks'ın 200 milyon kullanıcısının konum bilgisi de dahil olmak üzere birçok kişisel bilgisi sızdırılmıştır [1]. Bu tip veri sızıntıları ve konum verilerinin doğası gereği kişiler hakkında izinsiz çıkarımlarda bulunmak amacıyla kullanılabilen bir öznitelik olması, birçok insanı konum mahremiyeti konusunda endişelendirmektedir. Öyle ki, kullanıcılar çeşitli durumlarda KTS tarafından sunulan servisten aldıkları faydanın düşmesi pahasına konum mahremiyeti talep edebilmektedirler.

Günümüzde sosyal ağların ve sağladıkları KTS'lerin kullanımının artmasıyla beraber, kişiler arasındaki ilişkiler daha rahat açığa çıkmaya başlamıştır. Birçok sosyal ağ, kişilerin konum bilgisiyle beraber kiminle olduğu bilgisine de erişebilmektedir. İnstagram'a konum etiketiyle birlikte arkadaşlarımızla olduğumuz bir fotoğraf yüklediğimizde, açıkça İnstagram hem konum bilgimize hem de kiminle olduğumuz bilgisine sahip olacaktır. Facebook Yakındaki Arkadaşlar servisi gibi yakınlık-tabanlı konum servislerini kullanan kişiler için, KTS hem kişilerin konum bilgisine hem de yörünge geçmişine bakarak hangi arkadaşlarıyla (veya yakınlarıyla) birlikte olduğu bilgisine ulaşabilir. Aynı şekilde, Swarm uygulamasında kişiler birlikte bulundukları arkadaşlarını etiketleyebildikleri için, şüphesiz ki Swarm kişilerin konumlarına ve kiminle birlikte oldukları bilgisine çok rahat erişebilecektir. Literatürde yörünge veritabanlarından ortak konumda bulunma ve ortak hareket bilgisini elde etmeye yönelik çalışmalar da mevcuttur [18, 49]. Ortak konumda bulunma bilgisinin erişilebilirliğinin

artması, kişilerin konum mahremiyetinin korunurken diğer kişilerden izole olarak değerlendirilemeyeceği sorununu da beraberinde getirir ve konum mahremiyetinin korunması konusunda ek önlemler almayı gerektirmektedir. Literatürde hem konum hem de ortak konumda bulunma bilgisini kullanan saldırıların oldukça güçlü olabileceğini gösteren çalışmalar mevcuttur [38, 47].

1.1 Tezin İçeriği

Bu tez çalışmasının bundan sonraki kısımları şu şekilde düzenlenmiştir: Bölüm 2'de mahremiyet problemi altbaşlıklara ayrılarak detaylı bir literatür taraması yapılmış, hassas veri gizleme problemi ve konum servislerinin farklı uygulamaları üzerinden oluşan mahremiyet problemleri incelenip bunlarla ilgili literatürdeki önemli çalışmalara ve çözüm yöntemlerine değinilmiştir. Bölüm 3'te ilk olarak şehir ağı üzerinde perdeleme haritası oluşturmak için efektif bir sezgisel yöntem ve ilgili hareketlilik modeli önerilmiştir. Sonrasında bu yöntem kullanılarak gerçek şehir ağı çizgelerinden veri setleri oluşturulmuş, konum k-anonimlik ilkesine göre anlık ve tarihsel konum mahremiyeti modelleri tanıtılmış ve kullanıcıların konum mahremiyeti korumalı konum güncellemeleri yapabilmesini sağlayan algoritmalar geliştirilip kapsamlı deneysel çalışmalar sunulmuştur. Bölüm 4'te KTS'nin kullanıcıların birlikte bulunma/hareket bilgisini sayısallaştırabileceği ve kullanıcıların birbirinden izole olarak değerlendirilmesinin yetersiz kalacağı ortak konumlandırma saldırıları üzerine çalışılmış, kullanıcıları bu tip saldırılardan korumak amacıyla Bölüm 3'te önerilen çatıya ek olarak yeni bir anlık konum mahremiyet modeli tanıtılmış ve kullanıcıların konum mahremiyeti korumalı konum güncellemesi yapabilmesini sağlayan algoritmalar geliştirilip kapsamlı bir deneysel çalışma sunulmuştur. Bölüm 5'te ise tez çalışmasından çıkan genel sonuçlar incelenmiş ve ileride yapılabilecek çalışmalardan bahsedilmiştir.

2. İLGİLİ ÇALIŞMALAR

Günümüzde kullanıcı verileri, şirketler için çok değerli bir kavram olmuş durumdadır ve birçok uygulamanın tasarımında ve geliştirilmesinde önemli rol oynamaktadır. Makine öğrenmesi ve büyük veri tekniklerindeki ilerlemeyle beraber, birçok şirket iş modellerinin bir parçası olarak kullanıcılarından daha çok veri toplamayı amaçlamaktadır ve kullanıcılarını kendi ekosistemlerinin içine çekerek kullanıcılarıyla ilgili çok fazla veri toplamaktadırlar. Birçok ülkede kullanıcı verilerinin tutulması ve paylaşılması, mahremiyet ile ilgili yasalarla regüle edilse bile, ticari kaygılarla bu verilerin kullanılması yada bu verilerin dışarıdan gelen saldırılarla istemsizce sızdırılarak herkes tarafından erişilebilir olması, birçok insanı güvenlik ve mahremiyet konusunda daha duyarlı hale getirmiş durumdadır.

Mahremiyet konusu, literatürde farklı ayarlar üzerinde detaylıca çalışılmıştır. Bu tez çalışması kapsamında yapılan literatür taraması, bölümün devamında altbaşlıklara ayrılarak ve konuyla ilgili önemli çalışmalardan bahsedilerek okuyucuya sunulmaktadır.

2.1 Veri Mahremiyetine Yönelik Çalışmalar

Kullanıcı verileri, bir çok şirket, uygulama yada servis tarafından farklı amaçlarla saklanmakta ve kullanılmaktadır. Kullanıcılar, şirketlerle hassas verilerini paylaşmamayı veya hassas verilerine erişmek isteyen bir uygulamayı kullanmamayı seçebilir. Fakat hassas bilgiler paylaşılmasa bile, veri madenciliği tekniklerinin mevcut verilerden istatistiksel sonuçlar çıkararak hassas verileri yeniden oluşturmak için kullanılabileceği gösterilmiştir [15]. Dolayısıyla veri madenciliği teknikleri, veritabanı güvenliği için bir tehdit oluşturmaktadır. Bir kişinin hangi verisinin hassas/mahrem olduğu, kişiden kişiye değişkenlik gösterebilir veya kullanıcı verilerine sahip şirketler, hangi verilerin mahrem olup olmadığı konusunda farklı tutumlar sergileyebilir. Bu tip durumlarda veritabanı/veriseti yayınlanması esnasında mahremiyetin korunmasına yönelik literatürdeki genel yaklaşım, hassas verilerin önceden tanımlanmasını içerir. Daha sonra bu tanımlanan hassas veriler, veritabanından hassas kural tanımlarına göre filtrelenebilir [28]. Bir başka yöntemde, veriseti üzerinde hassas verileri veritabanından gizlemenin bir yolu olarak veri karıştırma yöntemi önerilmiştir [6]. Bu yöntemde, hassas veriler bir rastgele fonksiyon kullanılarak değişime uğratılır ve sonuçta oluşan

veritabanındaki veri dağılımı, orijinal veritabanındaki veritabanından farklı olur. Ayrıca değişime uğratılan veritabanının bireysel veri kayıtlarındaki orijinal değerleri doğru bir şekilde tahmin etmek mümkün olmasa da, orijinal veri dağılımını doğru bir şekilde tahmin etmek için yeni bir yeniden yapılandırma prosedürü önerilmiştir. Fakat hassas verileri gizlemeye çalışmak, beraberinde bir başka problemi yaratmıştır. Mahremiyeti sağlamak amacıyla hassas veriler gizlenirken, veritabanından alınan fayda azalmaktadır ve alınan faydayı maksimize etmenin NP-Zor olduğu gösterilmiştir [28].

Literatürde hassas veri gizlemeyle ilgili birçok çalışmaya temel olarak kabul edilen klasik k-anonimlik modeli ise [26], yayınlanan veriseti içindeki anahtar olarak kullanılmayan verilerin genelleştirilmesine dayanır. Genelleştirilmeye ulaşmak için, her bir bilgi en az k-1 tane başka bilgiyle eşleştirilir. Bu yöntem, daha sonra her bir bilginin öznitelikleri oluşturulup kümelenerek geliştirilmiştir [5]. Önerilen yöntem sayesinde her bir bilgiyi, ilgili özniteliklerinin özelliklerine göre veriseti içindeki en az k-1 tane başka bilgiden ayırt etmek mümkün olmamaktadır.

K-anonimlik tabanlı çözüm yöntemleri, yayınlanan veritabanındaki bilginin çeşitliliği az olduğunda yada kümelemeyle ilgili önceden ele geçirilen bilgilerin mevcudiyeti durumunda yetersiz kalmaktadır. Buna bir çözüm olarak literatürde l-çeşitlilik modeli önerilmiştir [3]. L-çeşitlilik modeli, anonimleştirme mekanizmasındaki hassas değerler için anonimlik grupları içine çeşitlilik mekanizmasını ekler ve anonimlik gruplarındaki hassas verilerin en az l tane iyi temsil eden özniteliğe sahip olmasını gerektirir. Bununla birlikte, l-çeşitlilik modeli hassas verilerin özniteliklerinin hassasiyet değerlerinin birbirinde farklı olması durumunda bilgi sızıntısına neden olabilir. Örneğin, bir hassas verinin özniteliklerinden birinin değerinin pozitif olması, negatif olmasına göre daha fazla bilgi sızıntısına yol açabilir. Bu gibi durumlar için, l-çeşitlilik modelinin bir uzantısı olarak t-yakınlık modeli önerilmiştir [36]. Bu model, herhangi bir anonimlik sınıfındaki hassas bir özniteliğin dağılımı ve bu özniteliğin verisetindeki dağılımı arasındaki farkın en fazla t kadar olmasını gerektirir ve l-çeşitlilik modelinin yaratabileceği bilgi sızıntılarının engellemesi amacıyla kullanılabilir.

2.2 Hareketlilik Uygulamaları ile Konum Paylaşımında Mahremiyet

Uygulamanın tipine göre KTS'nin hizmet şekli değişkenlik gösterebilse bile, bu tez çalışmasında çalışılan problemin uygulama alanı olan hareketlilik uygulamaları kapsamında veri-merkezli ve servis-merkezli olmak üzere iki şekilde hizmet vermektedirler [11]. Veri-merkezli hareketlilik uygulamaları, kullanıcıların konum güncellemelerini saklamak/kaydetmek amacıyla hareketlilik verilerini toplar. Bunu çevrimdışı olarak (yani kullanıcılar konum verilerini belirli bir süre yerel olarak

saklayıp tüm yörüngeyi tek seferde paylaşırlar) veya çevrimiçi olarak (yani kullanıcılar konum verilerini her konum güncellemesini anında paylaşırlar) yapabilirler. Örneğin 2019 yılında başlayan ve günümüzde halen devam etmekte olan Covid19 salgınını kontrol etmek amacıyla, devletler veri-merkezli KTS'lerden de faydalanmaktadır [37]. Servis-merkezli hareketlilik uygulamalarında ise konum paylaşımı servis talebini cevaplayabilmek için bir gerekliliktir. Yani servis-merkezli hareketlilik uygulamaları çevrimiçi olarak (bir istek-yanıt mekanizması ile) çalışmaktadır. Navigasyon uygulamaları (Google Haritalar gibi), servis-merkezli hareketlilik uygulamalarının en yaygın örneğidir.

Hareketlilik uygulamaları ile yapılan konum paylaşımlarında, *kimlik* ve *konum* mahremiyeti olarak iki tip mahremiyet sorunu ortaya çıkmaktadır. Kimlik mahremiyeti sorununda kullanıcıların hareketlilik verileri, bir grup kullanıcı içinden ayırt edilmelerini sağlabilecek bir öznitelik olarak kullanılabilir [40]. Kullanıcılar kimlik mahremiyetinin sağlanması adına bir insan topluluğu içinde anonim olmayı talep ederler. Konum mahremiyeti sorununda ise kullanıcılar KTS'ye kayıtlıdır ve kimlikleri bilinmektedir. Bu sebeple kullanıcılar izole olarak değerlendirilir ve konum mahremiyetinin sağlanması adına (örneğin izlenmemek için), bir grup konum içinde (örneğin perdelenmiş bir bölge [8]) anonim olmayı talep ederler. Bölümün devamında, literatürde hareketlilik uygulamaları kapsamında kimlik ve konum mahremiyetiyle ilgili yapılan çalışmalardan bahsedilmektedir.

2.2.1 Kimlik mahremiyetine yönelik çalışmalar

Hareket (yörünge) bilgisi, kimliklerin açığa çıkarılmasında kullanılabilen güçlü bir özniteliktir. Örneğin, veri-merkezli hareketlilik uygulamalarında yörünge veritabanlarının çevrimdışı olarak yayınlanması sırasında kullanıcıların kimlikleri açığa çıkarılabilir [40]. Bu tip durumlarda kullanıcılar kesin konumlarını KTS ile paylaşır ve KTS bu biriken yörünge verisetlerini üçüncü taraflarla paylaşmadan önce, yörüngelerden anonimlik grupları yaratıp her bir anonimlik grubu içindeki yörüngelerin diğer yörüngelerden ayırt edilememesini sağlayarak anonimleştirir. Kullanıcı kimlikleri, çevrimiçi yörünge veritabanı yayınlanmasında da açığa çıkarılabilir [55]. Bu tip durumlarda ise, çevrimdışı aşamada mevcut yörünge veritabanı incelenerek hassas hareketlilik örüntüleri tanımlanır. Bu hassas örüntüler, çevrimiçi aşamada saldırganın arka plan bilgisi olarak değerlendirilir ve bildirilen konumlar, kullanıcı kimliklerinin açığa çıkarılamaması amacıyla çevrimiçi olarak gizlenir. Literatürde yörünge veritabanlarında kimlik bilgisini gizlemeyle ilgili başka çalışmalar da mevcuttur [29, 32, 54].

Anonimlik tabanlı yöntemlerden biri olan k-anonimlik ilkesi, kimlik mahremiyetini sağlayabilmek için her kullanıcı kaydının en az k-1 diğer kullanıcı kaydından ayırt edilemez olması gerektiğini belirtir [41]. Bu sayede hangi kaydın kime ait olduğunun bulunamaması amaçlanır. Konum k-anonimlik ilkesi ise [22], klasik k-anonimliğinin bir uzantısı olarak önerilmiş ve servis-merkezli hareketlilik uygulamalarında bir KTS talebinin aynı konumu aynı anda paylaşan en az k-1 diğer kullanıcının isteklerinden ayırt edilememesi gerektiğini belirtir. Bu amaçla, kullanıcı haritası (uzaysal dünya) dikdörtgensel parçalara bölünür ve KTS istekleri her bölgeden en az k farklı istek gelene kadar gecikmeye uğratılır. Bunu sağlayabilmek adına, kullanıcılar servis isteklerinin bir parçası olarak güvenilir bir üçüncü tarafa konumlarını gönderirler ve mahremiyet gereksinimlerini güvenilir üçüncü taraf sağlar. Kişilerin istenilen mahremiyet gereksinimleri de değişkenlik gösterebilir. Bu amaçla, her kullanıcı kendi anonimlik düzeyi olan k değerini belirleyebilir ve mahremiyet gereksinimleri kişiselleştirilebilir [9]. Ayrıca kişilerin yörünge geçmişi, uzay-zamansal korelasyonlar nedeniyle kişileri yeniden tanımlamak için kullanılabilir. Bu amaçla literatürde, anlık konum anonimliğin bir uzantısı olarak tarihsel konum anonimliği de önerilmiştir [10].

Konum k-anonimliği, bazı durumlar için yeterli olmayabilir. Örneğin aynı anonimlik grubu içindeki kullanıcılar, KTS'ye aynı türde sorgular (örneğin bana en yakın gece kulübünü bul) gönderebilir. Bu durumda sorgunun kendisi hassaslaşır ve kişileri yeniden tanımlamak amacıyla kullanılabilir. Bu tip durumlarda l-çeşitlilik ilkesi, KTS'lerde de mahremiyet sağlayabilmek için kullanılmıştır [19]. Ayrıca l-çeşitlilik ilkesi, literatürde anlık [33] ve sürekli [45] KTS sorguları kapsamında da incelenmiştir.

Kullanıcıların gerçek kimlikleri bilinmese bile, kullanıcılar izledikleri yollar ile ilişkilendirilebilir ve konum uygulamaları tarafından takip edilebilir. Bu amaçla literatürde karıştırma bölgesi kavramı tanıtılmıştır [4]. Bu çalışmada kullanıcılar, konum uygulamalarıyla iletişim kurmak için sahte kimlikler kullanır. Kullanıcıların uygulamalarla iletişimleri güvenilir bir üçüncü taraf tarafından sağlanır. Kullanıcılar kendileri için hassas olan bölgeleri (karıştırma bölgesi) önceden tanımlarlar. Sonrasında karıştırma bölgesine giren kullanıcıların sahte kimlikleri, izledikleri yolun takip edilememesi amacıyla güvenilir üçüncü taraf tarafından karıştırılır. Bu sayede, bu hassas bölgelere yapılan girişler ve çıkışlar arasında bir ilişki kurulamaması sağlanır ve kullanıcıların uzun vadeli hareketlerinin uygulamalar tarafından takip edilmesi engellenir. Karıştırma bölgesi kavramı, literatürde pek çok farklı ayarda detaylıca çalışılmıştır [7, 34, 35, 42, 51, 52].

2.2.2 Konum mahremiyetine yönelik çalışmalar

Kullanıcılar, güvenilir KTS'lerle gerçek konumlarını şüphe etmeden paylaşabilirler. Bu nedenle bu tezde odaklanılan konu, saldırgan KTS'lerle (veya diğer veri alıcılarıyle) yapılan konum paylaşımlarında konum mahremiyetinin nasıl sağlanacağıdır. Veri-merkezli hareketlilik uygulamalarında konum mahremiyetini sağlamayla ilgili literatürde sahte konumlar [23] ve konum perdeleme [8, 12, 22, 31] olmak üzere iki teknik kullanılmaktadır.

Sahte konumlar yönteminde, kullanıcı KTS'ye gerçek konumu da dahil olmak üzere bulunabileceği birden fazla konum gönderir [23]. Burada sahte konumlar, geçici olarak tutarlı olacak şekilde oluşturulur. KTS, gelen bütün konumlar için kullanıcıya cevap döndürür ve kullanıcı bu dönen cevap içerisinden ihtiyacı olanı filtreleyerek KTS'den faydalanır. KTS, bu konumlar içerisinden gerçek kullanıcı konumunu ayırt edemeyeceği için kullanıcının konum mahremiyeti sağlanmış olur.

Konum perdeleme yönteminde ise kullanıcılar gerçek konumları yerine gerçek konumlarını da içeren perdelenmiş bir bölgeyi KTS'yle paylaşırlar. Literatürde perdelenmiş bölgeleri oluşturma ile ilgili birçok farklı yöntem vardır. Yöntemden yönteme değişkenlik gösteren perdelenmiş bölgenin şekli, bir daire [8], bir dikdörtgen [31] veya bir altçizge [16] şeklinde olabilmektedir. Perdelenmiş bölge oluşturma süreci, perdelenmiş bölgelerin şeklinden bağımsızdır ve iki aşamada çalışır. İlk olarak çevrimdışı aşamada kullanıcıların konum mahremiyet profillerine göre perdeleme haritaları oluşturulur ve dağıtılır. Kullanıcıların konum mahremiyet profili, her bir perdelenmiş bölgedeki minimum farklı konum sayısı olarak tanımlanır. İkinci aşama olan çevrimiçi aşamada ise, gerçek kullanıcı konumları perdeleme haritasındaki ilgili perdelenmiş bölge ile eşlenir ve bu perdelenmiş bölge KTS ile kullanıcının konumu olarak paylaşılır. Her perdeleme haritası, ilgili kullanıcının konum mahremiyet profiline göre oluşturulduğundan dolayı, konum güncellemelerinde paylaşılan perdelenmiş bölgeler üzerinde kullanıcıların konum anonimliği sağlanmış olur. KTS'den alınan faydayı ayarlamak için ise perdelenmiş bölgelerin ayrıntı düzeyi uygun biçimde seçilebilir.

Sık gerçekleşen konum güncellemelerinde konum mahremiyetini koruma problemi, ara sıra gerçekleşen konum güncellemelerindekine göre daha zordur. Buradaki mahremiyet problemi, tarihsel yörünge geçmişi üzerindeki uzay-zamansal korelasyonlardan kaynaklanır [14, 21]. Örneğin, kentsel alan çizgelerini ve yol hız sınırlarını arka plan bilgisi olarak kullanan hız-tabanlı saldırıların konum mahremiyetini tehlikeye atma konusundaki etkinliğini gösteren pek çok çalışma mevcuttur [13, 16, 20]. Ayrıca konumların sahip olabileceği hassas anlamsal etiketlerin (örneğin gece klübü,

ibadethane) mevcudiyeti durumunda da kullanıcıların konum mahremiyeti tehlikeye atılabilir. Bu tip durumlarda, l-çeşitlilik ilkesi kullanılarak kullanıcıların konum mahremiyetini korumak amacıyla her perdelenmiş bölge içindeki anlamsal konumların çeşitliliği istenebilir [27, 31]. Fakat buradaki pratik zorluk, konumların sahip olduğu anlamsal etiketlerin yeterli düzeyde elde edilememesi ve her kullanıcının kendi duyarlılık düzeyini (örneğin hassas konumların ne düzeyde hassas olduğunu) ayarlaması gerekmesinden kaynaklanır. Bu nedenle, bu tez çalışmasında daha pratik ve nesnel bir yaklaşım olan konum k-anonimlik yaklaşımı kullanılmıştır. Ayrıca literatürde konum k-anonimliğini, en az k farklı kullanıcıdan gelen konum güncelleme isteklerinin karıştırılması olarak tanımlayan çalışmalardan [22] farklı olarak, bu tez çalışmasında konum k-anonimliği oluşturulan tüm perdelenmiş bölgelerde kullanıcının konum güncellemesi yapabileceği en az k farklı konumun olduğu durumu ifade eder.

Servis-merkezli hareketlilik uygulamalarında konum paylaşma durumu daha farklıdır ve kullanıcıların istediği servisin sağlanabilmesi için konum verileri bir gereklilik olmaktadır. Yani servis-merkezli konum uygulamaları, veri-merkezli konum uygulamalarının aksine bir istek-yanıt mekanizması ile çalışır. Fakat servis-merkezli KTS'lerin güzel bir özelliği, gerçek konuma yakın konumlarda çalıştırıldıklarında servis kalitesinin fazla düşmemesidir. Bu bağlamda, veri-merkezli konum uygulamaları için önerilmiş olan sahte konum [23], PROBE [31] veya Casper [12] gibi çalışmaların çevrimiçi varyantları, servis-merkezli durumlarda da kullanılabilir.

Literatürde konum mahremiyetini korumak amacıyla k-anonimlik tabanlı kriptografik yöntemler de geliştirilmiştir [50, 53]. Fakat bu tarz yöntemler, işlemsel maliyetlerinin yüksek olması sebebiyle verimli değildir ve işlem kapasitesi düşük olan mobil cihazlarda geniş çaplı uygulanabilirliği pratiklikten uzaktır.

3. KENTSEL ALAN HAREKETLİLİĞİNDE ANONİM KONUM PAYLAŞIMI

Literatürde önerilen konum perdeleme tekniklerinin, aktif olarak hizmet veren KTS'ler üzerinde uygulanmasına yönelik bazı pratik zorluklar vardır. Bu konum perdeleme tekniklerinin ana dağıtım mimarisi, konum paylaşımı başlamadan önce perdelenmiş bölgelerden oluşan bir perdeleme haritasının oluşturulmasını ve konum paylaşımı esnasında KTS'nin bu haritadan haberdar edilmesini gerektirir. PROBE [31] ve benzeri [24, 30] yaklaşımlar bu şekilde çalışır. Fakat bu tip mimariler, beraberinde yüksek işlemsel maliyet ve iletişim maliyeti getirebileceği için KTS'ler ve mobil cihaza sahip kullanıcıları tarafından benimsenmesi güçtür. Ayrıca, bu yaklaşımlarda perdelenmiş bölgeler oluşturulurken iki yer arasındaki en kısa yol, uzaysal boyutta düz bir çizgi olarak ölçülür. Fakat insan hareketlerinin şehir ağı üzerinde sınırlı olduğu kentsel alanlarda bu yaklaşım zar zor geçerlidir. PROBE'un kentsel alanlar için çalışıldığı ayarda [16] ise, en kısa mesafeler zaman boyutunda değerlendirilmiş ve kentsel alanlardaki insan hareketlerini daha iyi yansıtmaktadır. Bu bağlamda, bu tezde önerilen çatıda da bu yaklaşım benimsenmiştir. Fakat PROBE ve kentsel alan dengi, her bir kullanıcı profili için bilinçli girilmesi gereken çok fazla parametre gerektirir. Bu tarz parametrik çözümler de kullanışlı değildir ve kullanıcılara konum mahremiyeti sağlama konusunda fazladan yük getirmektedir.

Bu tez çalışmasının bu bölümünde, bu tip pratik zorluklar değerlendirilerek aktif KTS'ler tarafından uygulanabilirliği yüksek olan anonimlik tabanlı bir konum paylaşım çatısı önerilmektedir. Önerilen çatı, KTS'den alınan faydayı maksimize edecek efektif bir sezgisel perdeleme haritası oluşturma yöntemine sahiptir ve KTS ile kullanıcılar arasında herhangi bir perdeleme haritası değişimine gerek yoktur. Ayrıca önerilen çatı, kullanıcıların konum mahremiyetlerini korumak amacıyla zayıf (anlık) ve güçlü (tarihsel) konum k-anonimliği modellerine sahip olması yönünden de literatürde eşsizdir. Literatürde önerilen etkin konum perdeleme yöntemleri PROBE [31] ve Casper [12], çevrimiçi olarak çalışan servis-merkezli yöntemlerdir. Yani konum mahremiyeti ihlali yaratan konum güncelleme istekleri, ileriki bir zamanda gönderilecek şekilde zaman gecikmesine uğratılır. Fakat burada önerilen çatı, tamamen veri-merkezli bir yöntemdir ve konum mahremiyeti ihlali yaratan konum güncelleme istekleri engellenir. Sonuçta ortaya çıkan kullanıcı yörüngeleri, zaman ekseninde de perdelenmiş olur ve bu yörüngeler KTS ile çevrimiçi veya çevrimdışı olarak paylaşılabilir.

3.1 Problem Formülasyonu

Problem formülasyonu, (i) hareketlilik modeli, (ii) mahremiyet modelleri ve (iii) saldırı modellerinin detaylandırılmasını içerir. Hareketlilik modelinde kullanıcıların kentsel alanlardaki hareketliliği modellenir. Gerçek kullanıcı yörüngeleri, kentsel alanın çizge temsili olan ACN (*Annotated City Network*) üzerinde tanımlanır. Her kullanıcının gerçek kullanıcı yörüngesi hassas olduğu için, kaba kullanıcı yörüngelerini tanımlamak amacıyla ACN'in kaba bir çizge modeli olan ACCN (*Annotated Coarse City Network*)'ler üretilir. Kullanıcılar ACN'e göre tanımlanan ve hassas olan gerçek kullanıcı yörüngelerini gizli tutar ve ACCN üzerinde tanımlanan kaba kullanıcı yörüngelerini KTS (saldırgan) ile paylaşırlar. Mahremiyet modelinde konum k-anonimliğin uzantısı olan zayıf ve güçlü konum k-anonimlik kavramları tanıtılır. Zayıf konum k-anonimliği kullanıcılara anlık konum anonimliği sağlamak amacıyla tanıtılmaktadır. Güçlü konum k-anonimliği ise kullanıcılara tarihsel konum anonimliği sağlamak amacıyla tanıtılmaktadır. ACCN'in her bir düğümü, mahremiyet modeliyle eşleşmesi için ACN'in en az k düğüm içeren bir altçizgesine karşılık gelmektedir. Saldırı modelinde ACN, ACCN ve anonimlik değerleri (k) saldırganın arkaplan bilgisi sayılır. Saldırganın görevi, hız-tabanlı saldırılardan faydalanarak paylaşılan kaba kullanıcı konumlarından (ACCN'e göre tanımlanan) gerçek kullanıcı konumlarına (ACN'e göre tanımlanan) ilişkin çıkarımlarda bulunmaktır.

3.1.1 Hareketlilik modeli

Kentsel alanlardaki kullanıcı hareketleri, şehir yol ağı ve bu yollarla bağlantılı mekanlar (ör. etkinlik alanları, yaşam alanları) üzerinde sınırlıdır. Bu amaçla, kullanıcı hareketleri şehir yol ağının çizge temsili olan açıklamalı şehir ağı üzerinde tanımlanır.

Tanım 1 (Açıklamalı şehir ağı) : Açıklamalı şehir ağı, şehir yol ağının ağırlıklı yönlü çizge temsilidir ve G = (V, E, w) şeklinde 3 öğeli olarak tanımlanır, öyle ki:

- V çizge üzerindeki düğümler kümesidir. Bu düğümler yerleşkeleri, kavşak noktalarını ve mekanları ifade etmektedir.
- $E \subseteq VxV$ çizgedeki kenarlar kümesidir. Örneğin $(v_1, v_2) \in E$ kenarı, iki kavşak noktası arasındaki yol parçasını veya bir kavşak noktası ile bir yerleşke (mekan) arasındaki yol parçasını gösterir.
- $w : E \rightarrow R^+$ çizgedeki kenarların ağırlık fonksiyonudur ve her kenarın karşılık geldiği yolu tamamlamak için gereken süreyi gerçek dünyadaki araç seyahat süresi cinsinden ifade eder.

Kullanıcı hareketlerini modellemek amacıyla, her bir $u \in U$ kullanıcısının herhangi bir zamanda ACN'nin bir $v \in V$ düğümünde bulunduğu varsayımı yapılmıştır. Bu varsayım altında, kullanıcıların koordinat sistemi üzerinde bulunduğu nokta (x,y) ACN üzerindeki en yakın düğüme atanır. Bu yöntem, KTS'ye karşı düşük seviyeli bir konum perdeleme sağlayacağı için de kullanışlıdır. Ayrıca bu varsayım, ACN çözünürlüğü yüksek bir şekilde modellenebileceği için de kısıtlayıcı olmayacaktır. Bu bağlamda, kullanıcıların gerçek (kesin) konumları aşağıdaki gibi tanımlanır.

Tanım 2 (Gerçek kullanıcı yörüngesi) : Gerçek kullanıcı yörüngesi TT_u, bir u kullanıcısın ACN G üzerinde t zamanındaki konumunu veren bir fonksiyondur. TT_u : $T \rightarrow V$ şeklinde gösterilir ve T zaman alanıdır. Örneğin $TT_u(t) = p$, u kullanıcısının $t \in T$ anında p düğümünde bulunduğunu söyler.

Eğer kullanıcı t anında bir düğüm yerine gerçekte bir yol üzerinde bulunuyorsa yada ağda modellenmeyen bir konumda ise, raporlanacak gerçek kullanıcı konumunun o anki konumuna en yakın düğümde olduğu kabulü yapılır. Dolayısıyla kullanıcılar herhangi bir anda bir düğümdedir ve hareketleri düğümden düğüme atlama şeklindedir.

Güvenilmez KTS'lerle gerçek kullanıcı konumunun paylaşılması, elbette kullanıcıların konum mahremiyetinin ihlal edilmesine neden olacaktır. Bu nedenle kullanıcıların kendi mahremiyet profillerine uyacak şekilde oluşturulmuş perdeleme haritalarına ihtiyacı vardır ve konum paylaşımı esnasında gerçek konumlarına perdeleme haritası üzerinde karşılık gelen perdelenmiş bölgeyi KTS ile paylaşırlar. Bu bağlamda ve aşağıda tanımlandığı üzere, ACN anonimlik grupları yaratmak amacıyla k-üyeli bölütlenir ve her bölütün en az k düğüme sahip olması sağlanır.

Tanım 3 (k-üyeli bölütlenmiş açıklamalı şehir ağı) : Bir k-üyeli bölütlenmiş açıklamalı şehir ağı (kPACN), ACN G = (V, E, w)'nin her bölütü en az k düğüme sahip olacak şekilde bölütlenmiş halidir. $G_k = (V = \{P_1, P_2, ..., P_m\}, E, w)$ şeklinde gösterilir ve m toplam bölüt sayısını ifade eder.

Her bir P_i bölütü, V üzerinde bir düğüm kümesi tanımladığı için, her bölütdeki düğümlerden bir tanesi ($p_i \in P_i$) ilgili bölütü tanımlamak amacıyla bölüt prototipi olarak atanır.

Tanım 4 (Prototipli k-üyeli bölütlenmiş açıklamalı şehir ağı) : Prototipli k-üyeli bölütlenmiş açıklamalı şehir ağı (prototipli kPACN), m bölütlü bir kPACN $G_k = (V = \{P_1, P_2, ..., P_m\}, E, w)$'nin her bölütünün tanımlayıcısı olarak ilgili bölütden bir düğüme sahip olan kPACN'dir. $G_k^p = (V = \{P_1^{p_1}, P_2^{p_2}, ..., P_m^{p_m}\}, E, w)$ şeklinde ifade edilir ve her p_i düğümü, P_i bölütünün prototipi olarak adlandırılır.

Her bir bölüt için temsili bölüt prototipi seçmek kolay değildir. Bunun nedeni, aşağıdaki yardımcı teoremde de gösterildiği üzere, prototip seçiminin çok sayıda farklı yolu olmasından kaynaklanmaktadır.

Yardımcı Teorem 1 (Prototipli k-üyeli bölütlenmiş açıklamalı şehir ağı sayısı) : Herhangi bir $G_k = (V = \{P_1, P_2, ..., P_m\}, E, w)$ için, $|P_1|x|P_2| \cdots |P_m|$ adet farklı prototipli kPACN vardır.

İspat : Her bir P_i bölütünden bir prototip p_i'yi seçmenin $|P_i|$ adet farklı yolu vardır ve $i \neq j$ olduğunda herhangi bir p_i seçmek, herhangi bir $p_j \in P_j$ seçmeye ortogonaldir.

Yukarıda bahsedilen Tanım 3 ve Tanım 4, perdeleme haritası oluşturmanın ilk aşamalarını tanımlamaktadır. Bu aşamaların ardından ACN'in kaba bir versiyonu olan ve perdeleme haritası olarak kullanılan ACCN, aşağıdaki gibi tanımlanır.

Tanım 5 (Açıklamalı kaba şehir ağı) : ACN $G = (V, E, w)$ verildiğinde, prototipli bir kPACN $G_k^p = (V = P_1^{p_1}, P_2^{p_2}, .., P_m^{p_m}, E, w)$ için ilgili açıklamalı kaba şehir ağı (ACCN), $G_k^{'} = (V^{'}, E^{'}, ew^{'}, vw^{'})$ şeklinde dört öğeli olarak tanımlanır, öyle ki :

- $V^{'} = \{p_1, p_2, ..., p_m\}$.
- $E^{'} = \{e_{ij} = (p_i, p_j) : \exists e = (v_1, v_2) \in E$ *,öyle ki,* $v_1 \in P_i$ *ve* $v_2 \in P_j . \forall_{i,j \in \{1,2,...,m\}} i \neq j\}$.
- $ew^{'} : E^{'} \rightarrow R^+$ kenar ağırlığı fonksiyonudur. Öyle ki, $ew^{'}(e_{ij} = (p_i, p_j)) = EnKisaYol_G(p_i, p_j)$, ACN G üzerinde p_i düğümünden p_j düğümüne olan en kısa yolun uzunluğunu ifade eder.
- $vw^{'} : V^{'} \rightarrow R^+$ düğüm ağırlığı fonksiyonudur. Öyle ki, $vw^{'}(p_i) = max\{max\{EnKisaYol_G(p_i, v), EnKisaYol_G(v, p_i)\} : v \in P_i\}$, ACN G üzerinde p_i düğümünden $v \in P_i$ düğümüne olan en kısa yol uzunluklarının maksimum değerini ifade eder.

ACCN, kullanıcıların KTS ile konum paylaşımlarında kullandığı perdeleme haritasıdır. Tanım 5'te ifade edildiği üzere, her bir prototipli kPACN (G_k^p) için ilgili ACCN ($G_k^{'}$) özel olarak oluşturulur. Bu nedenle her prototipli kPACN, kendi ACCN'ini tanımlamaktadır ve kişiye (anonimlik parametresi k ya göre) özeldir.

Kullanıcıların gerçek (kesin) kullanıcı yörüngelerini KTS ile paylaşması konum mahremiyet ihlaline sebep olabileceği için, aşağıda tanımlandığı üzere kaba kullanıcı yörüngelerinin KTS ile paylaşılması amaçlanmaktadır.

Tanım 6 (Kaba kullanıcı yörüngesi) : Kaba kullanıcı yörüngesi CT_u, bir u kullanıcısının ACCN $G_k^{'}$ üzerinde t zamanındaki konumunu veren bir fonksiyondur. $CT_u : T \rightarrow V^{'}$ şeklinde gösterilir ve herhangi bir $t \in T$ zamanında $TT_u(t) = v \in P_i$ ise $CT_u(t) = p_i$'dir.

ACCN'in her bir düğümü, ilgili prototipli kPACN'in prototip düğümlerinden birine karşılık gelmektedir. Bu bağlamda, Tanım 6'ya göre bildirilen her bir kaba konum $CT_u(t) = p_i$, u kullanıcısının gerçek konumunun P_i bölütündeki düğümlerden birinde olduğunu belirtir. Açıkca görüleceği üzere, kaba konum bir perdeleme bölgesidir ve ACN'nin bir altçizgesine karşılık gelmektedir.

Hareketlilik modeli, her kullanıcının sahip olduğu prototipli bir kPACN'yi ve ilgili perdeleme haritası ACCN'yi içerir. Dikkat etmek gerekirse, her kullanıcının prototipli kPACN'si (dolayısıyla ACCN'si) ve ayrıca k değerleri farklı olabilir.

3.1.2 Mahremiyet modeli

KTS'ye kullanıcıların konum mahremiyet beklentileri farklı olabileceği için her kullanıcıya aşağıda tanımlandığı üzere konum mahremiyet profili tanımlanır.

Tanım 7 (Konum mahremiyet profili) : Bir $u \in U$ kullanıcısının konum mahremiyet profili, istenen anonimlik seviyesini belirten k_u anonimlik parametresidir.

Başka bir deyişle, bir $u \in U$ kullanıcısı kaba konumunu paylaştığında, gerçek konumunun en az k_u adet düğüm üzerinde belirsiz olmasını ister. Konum mahremiyet profili de bu ihtiyacı tanımlar. Farklı kullanıcılar anonimlik parametresi için farklı değerler seçebildiğinden dolayı, Tanım 7 her kullanıcı için konum mahremiyet profilinin kişiselleştirilmesine izin verir.

Çalışılan problem kapsamında KTS saldırgan olduğu ve kullanıcılar konumlarını kaba olarak paylaşacağı için, KTS kullanıcıların gerçek konumunu bilemez. Fakat KTS, paylaşılan kaba konumları kullanarak kullanıcıların gerçek konumunun nerede olduğuna ilişkin kanılar oluşturabilir. Aşağıdaki tanım, KTS'nin kullanıcıların gerçek konumuna ilişkin kanılarını modellemektedir.

Tanım 8 (Kanı) : Kanı fonksiyonu $B_u^\tau : T \rightarrow 2^V$, bir u kullanıcısının nerede olduğuna ilişkin KTS'nin şu anki bilgisini (τ zamanında) tanımlar. Örneğin $B_u^\tau(t)$, KTS'nin u kullanıcısının t zamanındaki olası konumlarına ilişkin şu anki zaman τ'daki kanısını tanımlar. Yani KTS, şu anda $TT_u(t) \in B_u^\tau(t)$ olduğuna inanır. Bu sebeple $B_u^\tau(\tau)$ notasyonu, u kullanıcısının şu anki konumuna ilişkin KTS'nin şu anki kanısını verir.

Kullanıcılar, konum mahremiyetlerinin korunması için saldırgan KTS'nin kanıları üzerinde anonim olmayı talep ederler. Bu bağlamda, zayıf konum k-anonimliği kullanıcıların anlık konum belirsizliğini sağlamayı amaçlamaktadır.

Tanım 9 (Zayıf konum k-anonimliği) : ACN $G = (V, E, w)$, şimdiki zaman τ ve bir u kullanıcısı için anonimlik parametresi k_u verildiğinde, eğer $|B_u^\tau(\tau)| \geq k_u$ ise u kullanıcısı τ zamanında konum k-anonimdir.

Kullanıcılardan ard arda konum güncellemeleri geldiği durumda ise, kullanıcıların tarihsel yörüngelerindeki (yani geçmiş konumları) zamansal korelasyonlar nedeniyle kullanıcıların şu anki konum anonimlikleri ihlal edilebilir.Bu bağlamda aşağıda tanımlanan güçlü konum k-anonimliği (Tanım 10), kullanıcıların tüm yörüngesi üzerindeki tarihsel konum belirsizliğini sağlamayı amaçlar.

Tanım 10 (Güçlü konum k-anonimliği) : ACN $G = (V, E, w)$, şimdiki zaman τ ve bir u kullanıcısı için anonimlik parametresi k_u verildiğinde, eğer $|B_u^\tau(t)| \geq k_u : \forall t \in [0...\tau]$ ise u kullanıcısı τ zamanında konum k-anonimdir.

3.1.3 Saldırı modeli

Saldırı modelinde KTS, her kullanıcının konum güncelleme isteği olarak raporladığı ACCN'e göre tanımlanmış olan kaba kullanıcı yörüngelerinden, ACN'e göre tanımlanmış ve hassas olan gerçek kullanıcı yörüngelerine ilişkin çıkarımlarda bulunmaya çalışır. Bu amaçla saldırgan, aşağıdaki tanımda verilen arka plan bilgisini kullanabilir.

Tanım 11 (Arka plan bilgisi) : Saldırganın arka plan bilgisi aşağıdakilerden oluşur :

- Açıklamalı şehir ağı $G = (V, E, w)$
- Her bir kullanıcı $u \in U$ için :
  - Anonimlik parametresi k_u
  - k-üyeli bölütlenmiş açıklamalı şehir ağı $G_{k_u} = (V = \{P_1, P_2, ..., P_{m_{k_u}}\}, E, w)$
  - Prototipli k-üyeli bölütlenmiş açıklamalı şehir ağı $G_{k_u}^p = (V = \{P_1^{p_1}, P_2^{p_2}, ..., P_{m_{k_u}}^{p_{k_u}}\}, E, w)$
  - Açıklamalı kaba şehir ağı $G_{k_u}^{'} = (V^{'}, E^{'}, ew^{'}, vw^{'})$

Saldırganın arkaplan bilgisinin bir parçası olan ACN (G), tüm kullanıcılar için aynıdır. Ancak k-üyeli bölütlenmiş açıklamalı şehir ağı G_{k_u}, prototipli k-üyeli bölütlenmiş

açıklamalı şehir ağı $G^p_{k_u}$ ve ACCN $G^{'}_{k_u}$, her bir $u \in U$ kullanıcısının anonimlik parametresi k_u ile parametrize edilmiştir ve konum mahremiyet profili farklı olan kullanıcılar için bunlar da dolayısıyla farklı olacaktır.

Hedef bir u kullanıcısına yapılan saldırı şu şekildedir. KTS, zayıf k-anonimlik durumunda u kullanıcısına ilişkin kanısını $|B^\tau_u(\tau)| < k_u$ olacak şekilde daraltmaya çalışır. Bu amaçla, saldırgan tipik olarak hız-tabanlı saldırılardan yararlanabilir [21]. Güçlü k-anonimlik durumunda ise, KTS herhangi bir $t \in [0...\tau]$ zamanı için u kullanıcısına ilişkin kanısını $|B^\tau_u(t)| < k_u$ olacak şekilde daraltmaya çalışır. Her iki durumda da saldırgan eğer $B^\tau_u(\tau) = TT_u(\tau)$'ye ulaşırsa, u kullanıcısının hassas olan şu anki gerçek konumu elbette net bir biçimde açığa çıkacaktır.

3.2 Konum Anonimleştirme Çatısı

Önerilen konum anonimleştirme çatısı, (i) *çevrimdışı* ve (ii) *çevrimiçi* olmak üzere 2 aşamadan oluşmaktadır. Çevrimdışı aşamada, kullanıcılar konum mahremiyet profilleri olan k_u değerlerini belirler. Sonrasında her kullanıcının k_u değerine göre ilgili k-üyeli bölütlenmiş açıklamalı şehir ağları, prototipli k-üyeli bölütlenmiş açıklamalı şehir ağları ve açıklamalı kaba şehir ağları oluşturulur. Bu kullanıcıya özgü ağlar güvenilir bir üçüncü taraf tarafından oluşturulabilir ve dağıtılabilir. Fakat bu durumun tez çalışması kapsamında bir önemi yoktur. Çevrimiçi aşamada kullanıcılar gerçek kullanıcı yörüngelerini gizli tutarak kaba kullanıcı yörüngelerini hesaplayıp KTS ile paylaşırlar. Fakat KTS, paylaşılan kaba konumları ve arkaplan bilgisini kullanarak kullanıcıların gerçek kullanıcı yörüngelerine ilişkin çıkarımlarda bulunabilir. Bu nedenle, çevrimiçi aşamada paylaşılan kaba kullanıcı yörüngelerinin zayıf/güçlü konum k-anonimliği ihlali ihtimaline karşı kontrol edilmesi ve gerekirse engellenmesi gerekir.

Bu bölümde iki tane araştırma problemine yönelik çözüm önerileri detaylı bir şekilde anlatılmaktadır. İlki, çevrimdışı aşamada ACCN'lerin nasıl oluşturulacağıyla ilgilidir. Buradaki problem NP-Zor bir problemdir. Çünkü hem k-üyeli bölütlemeyi elde etmenin çok sayıda farklı yolu vardır [17], hem de Yardımcı Teorem 1'den hatırlamak gerekirse prototip seçiminin de çok sayıda farklı yolu vardır. İkinci araştırma problemi ise, çevrimiçi aşamada zayıf/güçlü konum k-anonimlik özelliklerinin nasıl korunacağıyla ilgilidir.

Bu bölümde önerilen çatıda kullanıcılar arasında herhangi bir ilişki olmadığı için, tüm kullanıcılar izole olarak ele alınmıştır. Bu nedenle, önerilen çözüm yöntemleri $k \leftarrow k_u$ olan bir u kullanıcısı üzerinden detaylandırılmaktadır.

3.2.1 Çevrimdışı aşama

Bu bölümde açıklamalı kaba şehir ağının nasıl oluşturulacağına yönelik çözüm önerisi detaylandırılmaktadır. Bu bağlamda Algoritma 1, ACCN oluşturma sürecinin ana aşamalarını göstermektedir. Algoritmanın ilk aşaması olan *kUyeliBolutleme* fonksiyonu, girdi olarak ACN (G)'i alır ve çıktı olarak her bölütü en az k düğüme sahip olan ve toplam bölüt sayısı $\approx |V|/k$ olan k-üyeli bölütlenmiş açıklamalı şehir ağı (G_k)'nı verir. K-üyeli bölütleme problemi NP-Zor olduğundan dolayı, bu amaçla literatürde sık kullanılan bir yöntem seçilip kullanılmıştır. *PrototipSecimi* fonksiyonu, girdi olarak G_k'yı alır ve bölümün devamında detaylı bir şekilde anlatıldığı üzere, NP-Zor olan prototip seçimi problemi için özel bir çözüm sunarak G_k^p'yi verir. Son aşama olan *ACCNOlusturma* fonksiyonu, girdi olarak G_k^p'yi alır ve kişiye özel olan perdeleme haritası ACCN ($G_k^{'}$)'i oluşturur.

Girdi: ACN $G = (V, E, w)$, anonimlik parametresi k
Çıktı: kPACN $G_k = (V = \{P_1, P_2, \cdots, P_m\}, E, w)$
Çıktı: prototipli kPACN $G_k^p = (V = \{P_1^{p_1}, P_2^{p_2}, \cdots, P_m^{p_m}\}, E, w)$
Çıktı: ACN $G_k^{'} = (V^{'}, E^{'}, ew^{'}, vw^{'})$
1: $G_k \leftarrow kUyeliBolutleme(G, k)$
2: $G_k^p \leftarrow PrototipSecimi(G_k)$
3: $G_k^{'} \leftarrow ACCNOlusturma(G_k^p)$
4: **return** G_k, G_k^p and $G_k^{'}$
Algoritma 1: Açıklamalı kaba şehir ağı (ACCN) oluşturma süreci.

PrototipSecimi fonksiyonu, prototip düğümlerin seçimiyle ilgili sezgisel bir yöntem kullanır ve aşağıda tanımlandığı üzere, ACCN'in kompaktlığını minimize eder.

Tanım 12 (ACCN'in kompaktlığı) : Prototipli bir kPACN $G_k^p = (V = \{P_1^{p_1}, P_2^{p_2}, ..., P_m^{p_m}\}, E, w)$ ve ilişkili ACCN $G_k^{'} = (V^{'}, E^{'}, ew^{'}, vw^{'})$ verildiğinde, ACCN'nin kompaktlığı kenar ağırlıklarının ve düğüm ağırlıklarının toplamı şeklinde tanımlanır ve aşağıdaki gibi formülize edilir.

$$Kompaktlik(G_k^{'}) = \sum_{e^{'} \in E^{'}} ew^{'}(e^{'}) + \sum_{v^{'} \in V^{'}} vw^{'}(v^{'}) \tag{3.1}$$

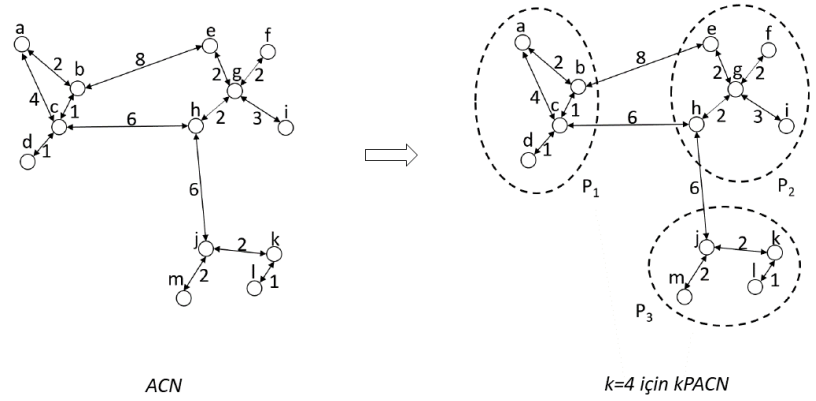
Dikkat etmek gerekirse, ACCN $G_k^{'}$ ne kadar kompakt olursa ACCN'in yol uzunlukları da o kadar küçük olacak ve bir sonraki bölümde gösterileceği üzere anonimlik ihlali riskleri de daha düşük olacaktır.

ACCNOlusturma fonksiyonu girdi olarak prototipli kPACN G_k^p'yi kullandığı için, ACCN'in kompaktlığı, prototipli kPACN G_k^p'de seçilen prototip düğümlere bağlıdır.

Fakat ACCN'in kompaktlığını minimize etmek için her bir prototip düğümü tek tek denemek pratiklikten uzaktır. Çünkü ACN'ler yüzbinlerce düğümden oluşabilen çizgelerdir ve Yardımcı Teorem 1'de gösterildiği üzere prototip seçiminin $|P_1|x|P_2|\cdots x|P_m|$ farklı yolu vardır. Bu amaçla *PrototipSecimi* fonksiyonu, MCLV (*Most Centrally Located Vertex*) sezgisel yöntemini kullanır. Sabit bir bölütleme yönteminde, $Kompaktlik(G_k^{'})$ tanımının ikinci bileşeni olan $\sum_{v^{'}\in V^{'}} vw^{'}(v^{'})$'nin minimize edilmesi için, her bölütdeki en merkezi konumdaki düğümün (MCLV) seçilmesi gerekir. Bu amaçla her bir P_i bölütündeki en merkezi konumdaki düğüm, aşağıdaki formülle basitçe hesaplanabilir.

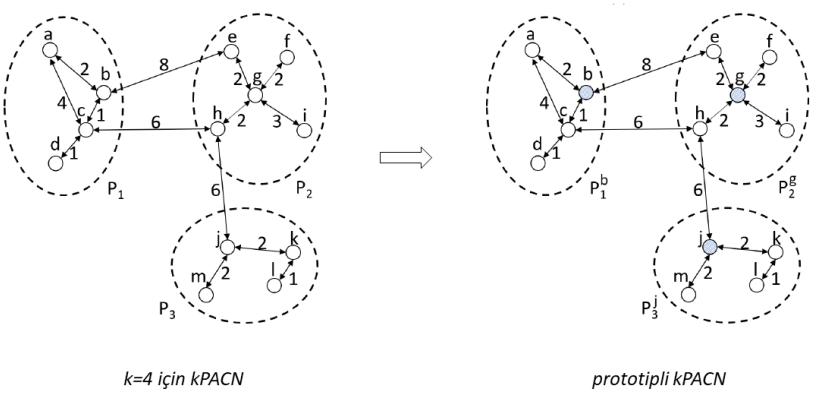
$$MCLV(P_i) = argmin_{p_i\in P_i} max\{EnKisaYol_G(p,p_i) : p \in P_i\} \tag{3.2}$$

Burada Algoritma 1'in her bir aşamasının çıktısı olan çizgeler saldırı modelindeki saldırganın arkaplan bilgisinin bir parçası olduğu için, *PrototipSecimi* fonksiyonunun çıktısı olan bütün çizgeler denktir ve MCLV sezgisel yaklaşımı ayrıca bir ifşa riski oluşturmaz.

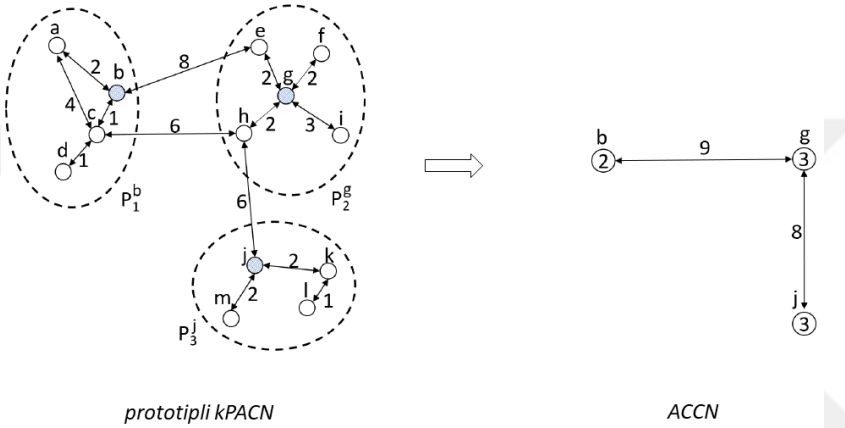


Şekil 3.1: K-üyeli bölütleme aşaması $G_k \leftarrow kUyeliBolutleme(G,k)$.

Örnek 1 (Algoritma 1'in çalışması): Bu örnekte Algoritma 1'in tüm aşamalarının nasıl çalıştığı gösterilmektedir. Şekil 3.1'de verilen örnek ACN (G)'yi ve $k = 4$'ü girdi olarak alan *kUyeliBolutleme* fonksiyonu, çıktı olarak kPACN'yi verir ve görüldüğü üzere her bölütünde en az 4 düğüm bulunan 3 bölütden oluşur. Ardından Şekil 3.2'de gösterildiği üzere *PrototipSecimi* fonksiyonu, kPACN'yi girdi olarak alır ve çıktı olarak prototipli kPACN'yi verir. Şekilde her bölüt için seçilen prototip düğüm gri gölgeli olarak görünmektedir ve MCLV sezgisel yöntemiyle seçilmiştir. Şekil 3.3'de gösterilen son aşamada *ACCNOlusturma* fonksiyonu, girdi olarak prototipli kPACN'yi alır ve çıktı olarak ACCN'i üretir. ACCN'in düğüm ağırlıkları, MCLV sezgisel yöntemine göre hesaplanır. Örneğin, *g* düğümü için düğüm ağırlığı MCLV sezgisel yöntemine göre 3 olarak hesaplanır. ACCN'in kenar ağırlıkları ise ACCN'e göre hesaplanır. Örneğin, *g*



Şekil 3.2: Prototip seçimi aşaması $G_k^p \leftarrow PrototipSecimi(G_k)$.



Şekil 3.3: ACCN oluşturma aşaması $G_k^{'} \leftarrow ACCNOlusturma(G_k^p)$.

düğümünden j düğümüne giden en kısa yol $\{g,h,j\}$ düğümleri üzerinden geçen yoldur ve ağırlıklarının toplamı 8'dir.

3.2.2 Çevrimiçi aşama

Bu bölümde çevrimiçi aşamada zayıf/güçlü konum k-anonimliğin (Tanım 9 ve Tanım 10'a göre) nasıl korunacağına ilişkin çözüm önerileri detaylandırılmaktadır. İlgili çözümler, her konum güncelleme isteğinde konum k-anonimlik özelliğinin zamana göre aşamalı olarak korunmasına dayanır. Mevcut konum güncelleme istekleri, hız-tabanlı saldırılara dayalı çıkarımlara karşı kontrol edilir ve konum k-anonimlik ihlali yaratan istekler engellenir.

Bir u kullanıcısının KTS ile ilk konum paylaşımı t_1 zamanında ve son konum paylaşımı $t_n \leq \tau$ zamanında olsun. Bu durumda u kullanıcısının konum paylaşımları (yani tarihsel yörüngesi), $t = \{t_1, t_2, \cdots, t_n\}$ zamanları için $(t, CT_u(t))$ şeklinde ikililerden oluşur. Bu bağlamda bölümün devamında kullanıcıların şu andaki zayıf ve güçlü konum k-anonimlik özelliklerinin nasıl korunacağına ilişkin ilgili problemler detaylandırmaktadır.

Problem 1 (Zayıf konum k-anonimliği). Şu anki zamanda $\tau > t_n$ bir u kullanıcısından konum güncelleme isteği geldiğini varsayalım. Tanım 9'a göre zayıf konum k-anonimlik özelliğini koruma problemi, bütün $t = \{t_1, t_2, \cdots, t_n\}$ zamanlarında $|B_u^t(t)| \geq k$ olacak şekilde $|B_u^\tau(\tau)| \geq k$ koşulunu sağlamaktır.

Problem 1'in ilk aşaması olan t_1 zamanını düşünelim. Burada, u kullanıcısı tarafından paylaşılan herhangi bir kaba konum $p_r \in V^{'}$, zayıf konum k-anonimlik özelliğini koruyacaktır. Çünkü, KTS'nin bu konum paylaşımına ilişkin kanısı $|B_u^{t_1}(t_1)| = P_r$ olacak ve tanım gereği herhangi bir P_r bölütü için $|P_r| \geq k$ koşulu zaten sağlanacaktır. Şimdi Problem 1'in son aşaması olan t_n zamanını düşünelim. Öyle ki, bu aşamada yapılan kaba konum paylaşımı $CT_u(t_n) = p_i$ olsun ve zayıf konum k-anonimlik özelliği $|B_u^{t_n}(t_n)| = |P_i| > k$ korunmuş olsun. Kullanıcının da şu anki kaba konumunun $CT_u(\tau) = p_j$ olduğunu varsayalım. Burada $i = j$ ve $i \neq j$ olarak iki ayrı senaryo incelenmelidir. Açıkca $i = j$ durumunda, kullanıcının $[t_n \cdots \tau]$ zaman aralığında hareketsiz kaldığı yada aynı bölüt içinde hareket ettiği KTS tarafından çıkarılamaz. Bu sebeple p_j konumunu paylaşmak, zayıf konum k-anonimlik özelliğini her zaman sağlar. Öte yandan, $i \neq j$ durumunda p_j konumunu paylaşmak, aşağıdaki teoremde açıklandığı üzere zayıf konum k-anonimlik ihlali riski yaratır. Aşağıdaki teorem aynı zamanda zayıf konum k-anonimlik özelliğinin zamana göre aşamalı olarak korumak için gerekli koşulu da sağlar.

Teorem 1 (Zayıf konum k-anonimlik özelliğini aşamalı olarak sağlama) Bir u kullanıcısının KTS ile paylaştığı en son kaba konumunun (t_n zamanında) p_i olduğunu ve KTS'nin buna ilişkin kanısının $B_u^{t_n}(t_n) = P_i$ olduğunu varsayalım. Kullanıcının şu andaki gerçek kullanıcı konumunun da $TT_u(\tau) \in P_j$ olduğunu varsayalım. Bu durumda, eğer $EnKisaYol_{G_k^{'}}(p_i, p_j) + vw^{'}(p_j) \leq \tau - t_n$ koşulu sağlanıyorsa, $B_u^\tau(\tau) = P_j$ (ve dolayısıyla $|B_u^\tau(\tau)| \geq k$) koşulu zaten sağlanacağı için u kullanıcısının şuandaki kaba konumunu $CT_u(\tau) = p_j$ olarak paylaşması güvenlidir.

İspat: Bir önceki paylaşılan konum (t_n zamanında) p_i'dir. P_j bölütü içinde p_i düğümünden herhangi bir başka düğüme gidilebilecek en kısa zaman
$EnKisaYol_{G_k^{'}}(p_i, p_j) + vw^{'}(p_j)$'dır. Eğer gerçek seyahat zamanı $\tau - t_n$,
$EnKisaYol_{G_k^{'}}(p_i, p_j) + vw^{'}(p_j)$'den küçük değil ise u kullanıcısı p_i düğümünden P_j bölütündeki herhangi bir düğüme ulaşabilir. Bu, zayıf konum k-anonimlik özelliğinin korunduğu ve $B_u^\tau(\tau) = P_j$ olduğu anlamına gelir. Fakat $ew^{'}(e_{ij} = (p_i, p_j)) + vw^{'}(p_j) > \tau - t_n$ olduğu durumda, P_j bölütündeki bazı düğümler ulaşılamaz hale gelir ve bu düğümler arka plan bilgisinden faydalanılarak filtrelenip zayıf konum k-anonimlik ihlali $|B_u^\tau(\tau)| < k$ olarak sonuçlanabilir. Bu nedenle, eğer
$EnKisaYol_{G_k^{'}}(p_i, p_j) + vw^{'}(p_j) > \tau - t_n$ olursa, u kullanıcısının konum güncelleme isteği engellenmelidir.

Problem 2 (Güçlü konum k-anonimliği). Şu anki zamanda $\tau > t_n$ bir u kullanıcısından konum güncelleme isteği geldiğini varsayalım. Tanım 10'a göre güçlü konum k-anonimlik özelliğini koruma problemi, bütün $t^{'} = \{t_1, t_2, \cdots, t_n\}$ zamanlarında $|B_u^{t_n}(t^{'})| > k$ olacak şekilde $|B_u^{\tau}(t)| > k$ koşulunu bütün $t = \{t_1, t_2, \cdots, t_n, \tau\}$ zamanları için sağlamaktır.

Problem 2'nin ilk aşaması olan t_1 zamanı, açık bir şekilde Problem 1'le aynıdır ve bu aşamada güçlü konum k-anonimlik özelliği her zaman korunur. Problem 2'in son aşaması olan t_n zamanını düşünelim. Öyle ki, bu aşamada yapılan kaba konum paylaşımı $CT_u(t_n) = p_i$ olsun ve güçlü konum k-anonimlik özelliği korunmuş olsun. Kullanıcının da şu anki kaba konumunun $CT_u(\tau) = p_j$ olduğunu varsayalım. Burada $i = j$ ve $i \neq j$ olarak iki ayrı senaryo incelenmelidir. Açıkca $i = j$ durumunda, kullanıcının $[t_n \cdots \tau]$ zaman aralığında hareketsiz kaldığı yada aynı bölüt içinde hareket ettiği KTS tarafından çıkarılamaz. Bu sebeple p_j konumunu paylaşmak, güçlü konum k-anonimlik özelliğini her zaman sağlar. Öte yandan, $i \neq j$ durumunda p_j konumunu paylaşmak, aşağıdaki teoremde açıklandığı üzere güçlü konum k-anonimlik ihlali riski yaratır. Aşağıdaki teorem aynı zamanda güçlü konum k-anonimlik özelliğinin zamana göre aşamalı olarak korumak için gerekli koşulu da sağlar.

Teorem 2 (Güçlü konum k-anonimlik özelliğini aşamalı olarak sağlama) Bir u kullanıcısının KTS ile paylaştığı en son kaba konumunun (t_n zamanında) p_i olduğunu ve KTS'nin buna ilişkin kanısının $B_u^{t_n}(t_n) = P_i$ olduğunu varsayalım. Kullanıcının şu andaki gerçek kullanıcı konumunun da $TT_u(\tau) \in P_j$ olduğunu varsayalım. Bu durumda, eğer $vw^{'}(p_i) + EnKisaYol_{G_k^{'}}(p_i, p_j) + vw^{'}(p_j) \leq \tau - t_n$ koşulu sağlanıyorsa, $B_u^{\tau}(\tau) = P_j$ (ve dolayısıyla $|B_u^{\tau}(\tau)| \geq k$) ve $B_u^{\tau}(t_n) = P_i$ (ve dolayısıyla $|B_u^{\tau}(t_n)| \geq k$) koşulları zaten sağlanacağı için u kullanıcısının şuandaki kaba konumunu $CT_u(\tau) = p_j$ olarak paylaşması güvenlidir.

İspat: KTS'nin bir önceki kanısı $B_u^{t_n}(t_n) = P_i$ verildiğinde, güçlü konum k-anonimlik ihlali olmaması için KTS'nin şu anki kanılarının (i) $B_u^{\tau}(\tau) = P_j$ ve (ii) $B_u^{\tau}(t_n) = P_i$ olması gerekir. Bu ancak P_i bölütü içindeki herhangi bir düğümden P_j bölütü içindeki herhangi bir düğüme seyahat süresi $\tau - t_n$'den az olmadığında mümkündür. Yani $\tau - t_n$ aralığındaki saldırı, P_i veya P_j bölütündeki herhangi bir düğümü filtrelemek için kullanılamamalıdır. ACCN ve prototipli kPACN'nin oluşturulma şekli sebebiyle, P_i ve P_j bölütlerindeki herhangi iki düğüm arasındaki minimum seyahat süresi $vw^{'}(p_i) + EnKisaYol_{G_k^{'}}(p_i, p_j) + vw^{'}(p_j)$'dir. Bu nedenle, eğer
$vw^{'}(p_i) + EnKisaYol_{G_k^{'}}(p_i, p_j) + vw^{'}(p_j) > \tau - t_n$ olursa, u kullanıcısının konum güncelleme isteği engellenmelidir.

Algoritma 2, Teorem 2'ye göre bir u kullanıcısı için güçlü konum k-anonimlik özelliğini sağlar. Algoritma, bir u kullanıcısından gelen konum güncelleme isteğini KTS'nin

Girdi: ACN $G = (V, E, w)$
Girdi: kPACN $G_k = (V = \{P_1, P_2, \cdots, P_m\}, E, w)$
Girdi: prototipli kPACN $G_k^p = (V = \{P_1^{p_1}, P_2^{p_2}, \cdots, P_m^{p_m}\}, E, w)$
Girdi: ACCN $G_k^{'} = (V^{'}, E^{'}, ew^{'}, vw^{'})$
Girdi: (Eğer mevcutsa) Bir önceki zaman t ve bu zamandaki kanı $B_u^t(t) = P_i$
Girdi: Şu anki zaman τ, $TT_u(\tau) \in P_j$
Çıktı: p_j veya **null**

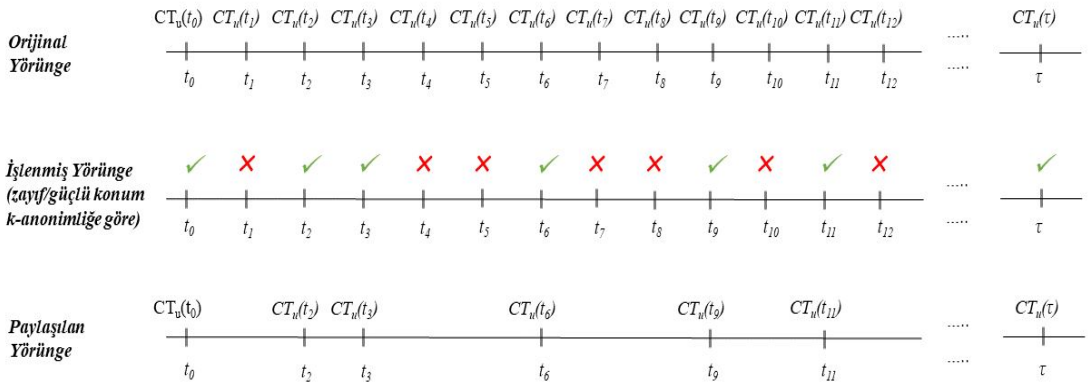
1: **if** $\tau = t_1$ **then**
2: // ilk konum paylaşımı
3: $t \leftarrow \tau$
4: $B_u^{t_1}(t_1) = P_j$
5: **return** p_j (kullanıcı bu konumu KTS ile paylaşır)
6: **else**
7: $anytoanydist \leftarrow vw^{'}(p_i) + EnKisaYol_{G_k^{'}}(p_i, p_j) + vw^{'}(p_j)$
8: **if** $anytoanydist \leq \tau - t$ **then**
9: $t \leftarrow \tau$
10: $B_u^t(t) = P_j$
11: **return** p_j (kullanıcı bu konumu KTS ile paylaşır)
12: **else**
13: **return** **null** (konum paylaşma isteği engellenir)
14: **end if**
15: **end if**

Algoritma 2: Anonimliği ihlal eden konum güncelleme isteklerininin seçici olarak engellenmesiyle güçlü konum k-anonimlik özelliğinin aşamalı olarak sağlanması.

kanılarını simüle ederek güçlü konum k-anonimlik ihlali riskine karşı kontrol eder. Algoritma eğer p_j döndürürse, bu konum güçlü konum k-anonimlik özelliğini sağlar ve kullanıcı bu konumu güvenle KTS ile paylaşabilir. Öte yandan, eğer algoritma **null** döndürürse, bu konum paylaşımı güvenli değildir ve engellenir. Bu nedenle, algoritma bazı konum güncellemelerini seçici olarak engelleyebilir. Bunun sonucu olarak, ortaya çıkan kaba kullanıcı yörüngeleri (perdelenmiş konumlar) zaman ekseninde de perdelenmiş olur.

Açıkca görüleceği üzere, Algoritma 2'nin 7. satırındaki uzaklık ifadesi $(vw^{'}(p_i) + EnKisaYol_{G_k^{'}}(p_i, p_j) + vw^{'}(p_j))$'nin yerine $(EnKisaYol_{G_k^{'}}(p_i, p_j) + vw^{'}(p_j))$ ifadesi kullanılarak Teorem 1'e göre bir u kullanıcısı için zayıf konum k-anonimliği koruyan algoritma elde edilebilir. Her iki algoritmanın güzel tarafı, $O(1)$ zamanda çalışmalarıdır. Bu çalışma zamanını elde edebilmek için, ACN G_k'ya göre daha küçük bir çizge olan ACCN $G_k^{'}$ üzerindeki en kısa yolların önceden hesaplanması ve çizge bölütlerinin hash-map şeklinde uygulanması gerekmektedir. Bu durumda, $vw^{'}(.)$'yi bulmak için beklenen zaman $O(1)$ olacaktır.

Şekil 3.4, çevrimiçi aşamada önerilen algoritmaların bir u kullanıcısının tüm yörüngesi üzerinde gerçekleşimini göstermektedir. Şekildeki orijinal yörünge, ilk konum paylaşımı



Şekil 3.4: Önerilen algoritmaların tarihsel yörünge üzerinde gerçekleşimi.

zamanı olan t_0'dan başlamakta ve son konum paylaşımı zamanı olan şu anki zaman τ'ya kadar olan kaba konum güncellemelerinden oluşmaktadır. Buradaki kaba konumlar, çevrimdışı aşamada (Bölüm 3.2.1) önerilen algoritma sonucunda u kullanıcısı için oluşturulmuş olan ACCN'e göre tanımlanan kaba konumlardır. Kaba konumlar, KTS ile paylaşılmadan önce kullanıcı tarafından bu bölümde önerilen algoritmalarla kullanıcının zayıf veya güçlü k-anonimlik durumundan hangisini tercih ettiğine göre işlenir ve ilgili k-anonimlik ihlali yaratan kaba konum paylaşımları engellenir. Sonuç olarak, bu örnek için KTS ile paylaşılan yörünge şekilde gözüktüğü gibi olacaktır. Mahremiyet ihlali yaratan kaba konumların engellenmesi, u kullanıcısının tarihsel yörüngesinin zaman ekseni üzerinde de perdelenmesine sebep olur ve bu durumun zamansal korelasyonları kullanan hız-tabanlı saldırılara karşı koruma sağlayacağı deneysel çalışmalarla da gösterilmiştir.

3.3 Deneysel Çalışmalar

3.3.1 Deneysel düzenek

Verisetlerinin elde edilmesi ve işlenmesi. Deneysel çalışmalarda kullanmak amacıyla *MustafaKemal*, *Osmaniye* ve *Ankara* olmak üzere 3 farklı boyutta gerçek ACN üretilmiştir. İlgili ACN'lerin kenar ve düğüm bilgileri OpenStreetMap [39]'den elde edilmiştir. Aynı zamanda kenarlar üzerindeki araç seyahat hız limitleri de yine [39]'den elde edilmiştir. Kenar ağırlıkları, her bir kenar için ilgili kenarın uzunluğunun ilgili kenar üzerindeki araç hız limitine bölünmesiyle seyahat süresi olarak elde edilmiştir ve ACN'lerin kenar ağırlıkları olarak bu seyahat süreleri kullanılmıştır. Araç hız limiti

olmayan kenarlar için hız limitinin basitçe 50 km/saat olduğu varsayılmıştır. Aşağıdaki çizelgede ilgili ACN'lerin bazı istatistikleri gösterilmektedir.

Çizelge 3.1: Verisetlerinin özellikleri.

Veriseti	ACN düğüm sayısı	ACN kenar sayısı
MustafaKemal	365	977
Osmaniye	11.571	31.115
Ankara	82.946	226.710

Algoritma 1'deki *kUyeliBolutleme* fonksiyonunu uygulamak için METIS [25] kullanılmıştır. Sonrasında k-üyeli bölütlerle ilişkili prototipli kPACN'ler ve ACCN'ler, MCLV sezgisel yöntemi kullanılarak elde edilmiştir. Şekil 3.5'de ACN'ler ve ilgili ACCN'lerin harita düzenleri gösterilmektedir. Her ACCN, ilgili k değeri kullanılarak oluşturulmuştur.

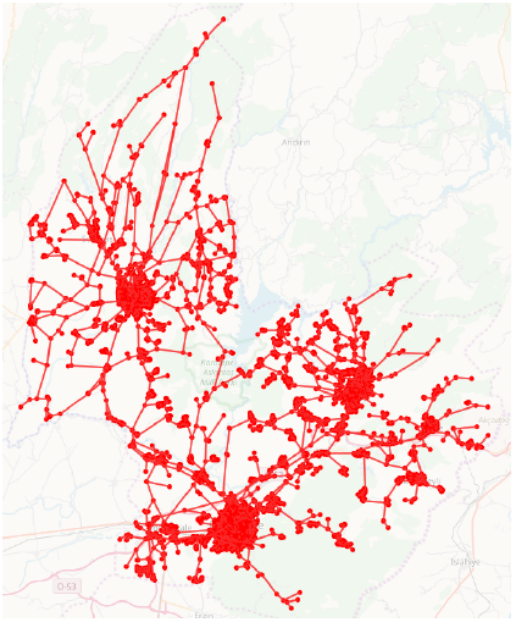
KTS'ye erişim simülasyonu. Bir u kullanıcısının KTS'ye erişimlerini simüle etmek amacıyla, her ACN üzerinde ilgili kullanıcı için sentetik kullanıcı yörüngeleri oluşturulmuştur. Yörünge oluşturma süreci şu şekildedir. Başlangıçta kullanıcı ilgili ACN üzerinde rastgele bir düğüme atanır. Ardından her bir iterasyonda, ilgili ACN üzerinde kullanıcının gidebileceği rastgele bir düğüm seçilir. Kullanıcının şuanda bulunduğu düğümle gideceği düğüm arasındaki mesafenin ew olduğunu varsayarsak, burada kullanıcıya ew ve $2 * ew$ arasında gerçek bir sayı seyahat süresi olarak atanır. Kullanıcının tüm uzay-zamansal yörüngesini elde etmek için bu süreç tekrar edilir. Deneysel çalışmalar kapsamında oluşturulan yörüngelerin uzunlukları *MustafaKemal* için 10000, *Osmaniye* için 200000 ve *Ankara* için 300000 olarak ayarlanmıştır.

3.3.2 Deneysel sonuçlar

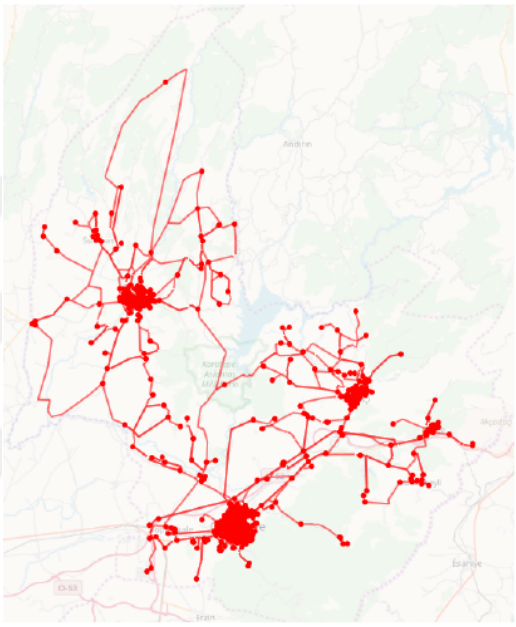
Çevrimdışı aşama (Bölüm 3.2.1)'nın algoritmaları Python diliyle, çevrimiçi aşama (Bölüm 3.2.2)'nın algoritmaları ise Java dili ile kodlanmıştır. Çevrimdışı hesaplamalar genelde ön işleme olarak kabul edildiği için çalışma zamanı üzerine bir etkisi olmaz. Çevrimiçi aşama algoritmaları ise her konum güncelleme isteğinde sabit zamanda çalıştığı için bu algoritmaların çalışma zaman verimliliği ölçülmeyip sadece algoritmaların efektifliği ölçülmüştür. Ayrıca bu tezde önerilen algoritmaların efektifliği, literatürdeki algoritmalarla kıyaslanamaz durumdadır Bunun sebebi, literatürdeki konum perdeleme algoritmalarının etkin bir şekilde servis-merkezli olmasından kaynaklanmaktadır, yani bu algoritmalar KTS talepleri her durumda yanıtlanacak şekilde konum güncelleme isteklerine zaman gecikmesi uygulamaktadırlar. Fakat bu tezdeki kurgu tamamen veri-merkezlidir ve güvenli olmayan konum güncelleme istekleri engellenir.

(a) *MustafaKemal* için ACN

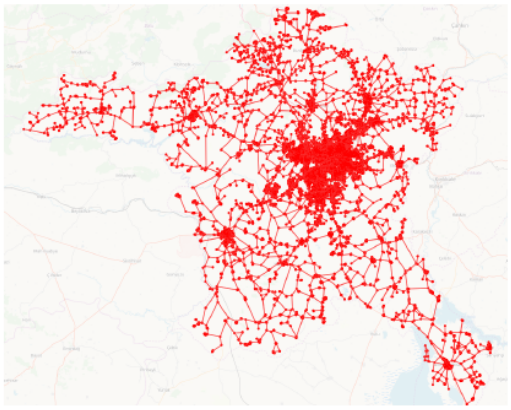
(b) *MustafaKemal* için ACCN (k=10)



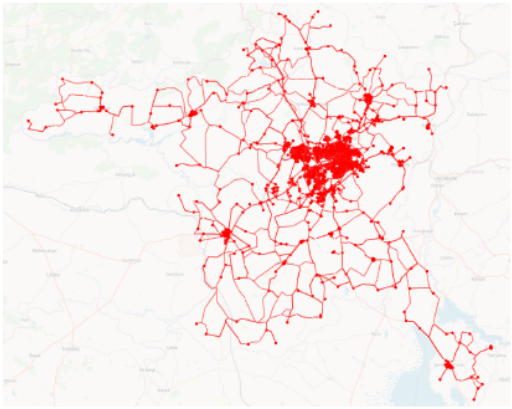
(c) *Osmaniye* için ACN



(d) *Osmaniye* için ACCN (k=20)

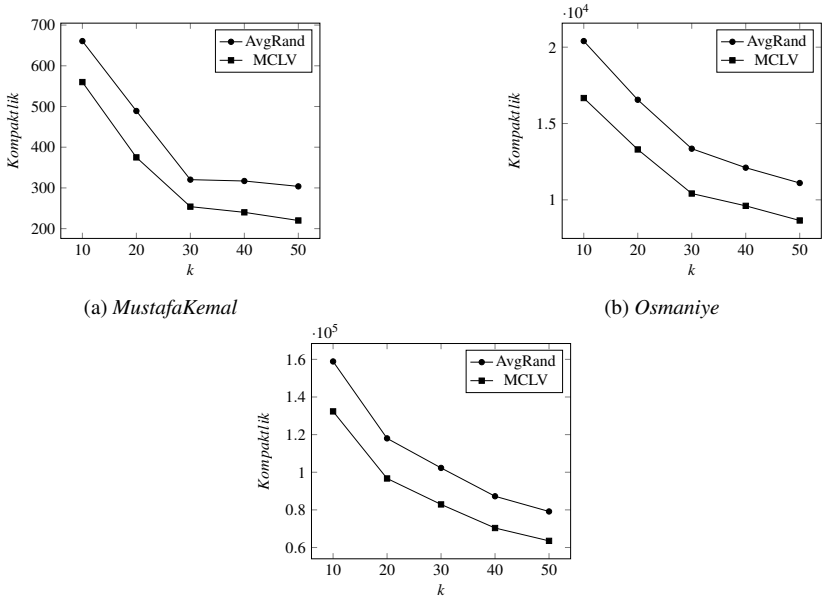


(e) *Ankara* için ACN



(f) *Ankara* için ACCN (k=50)

Şekil 3.5: Üç ACN'nin ve ACCN'lerinin harita düzenleri.



(a) *MustafaKemal* (b) *Osmaniye*

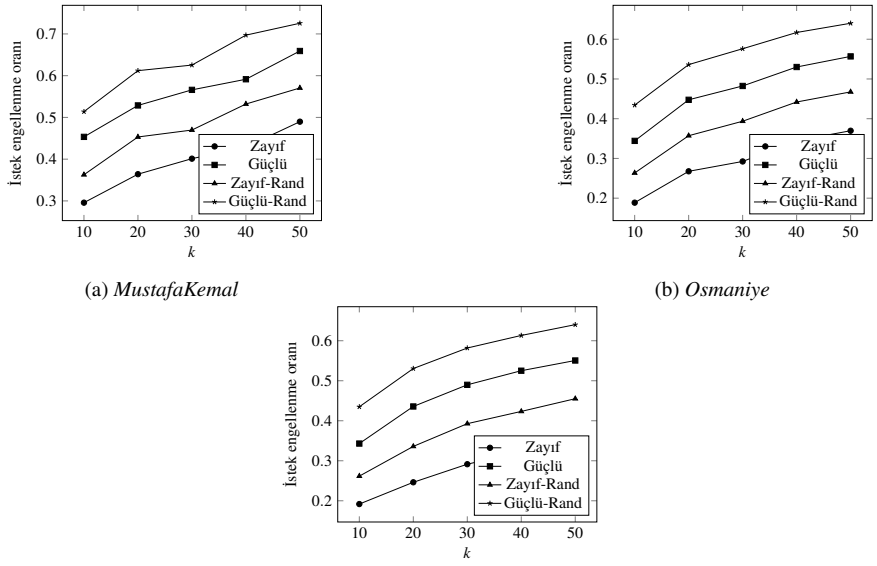
(c) *Ankara*

Şekil 3.6: Tanım 12'de tanımlanan ACCN kompaktlığı. *AvgRand*, 10 rastgele prototip seçiminin ortalamasıdır.

Çevrimdışı aşama Çevrimdışı aşamanın efektifliği Tanım 12'de verilen *Kompaktlik* fonksiyonu ile ölçülmektedir. *kUyeliBolutleme* fonksiyonunu uygulamak için METIS [25] kullanıldığından dolayı, *Kompaktlik* fonksiyonu burada *kUyeliBolutleme* fonksiyonunun (ve dolayısıyla MCLV sezgisel yönteminin) performansını ölçmektedir. Şekil 3.6'te 10 tane rastgele prototip seçiminin ortalaması ve MCLV sezgisel yöntemini karşılaştırılmaktadır. Sonuçlar, tüm verisetleri için MCLV sezgisel yönteminin rastgele prototip seçimine göre daha faydalı olduğunu doğrulamaktadır. Ayrıca, MCLV yöntemiyle üretilen ACCN'ler üzerinde 0.05 hassasiyet düzeyinde tek kuyruklu t-testi uygulanmış ve sonuçlar istatistiksel olarak da anlamlı bulunmuştur.

Çevrimiçi aşama Çevrimiçi aşama algoritmalarının (Algoritma 2 ve zayıf konum k-anonimlik varyantının) efektifliğini engellenen konum günceleme isteklerinin oranı ölçmektedir. Engellenen konum güncelleme isteklerinin oranı ne kadar düşük olursa, paylaşılan kaba konumun faydası da o kadar yüksek olacaktır.

Şekil 3.7, farklı k değerlerinde engellenen konum güncelleme isteklerinin oranını göstermektedir. Şekildeki *Weak-Rand* ve *Strong-Rand* grafikleri, çevrimdışı aşamanın sonuçlarında tartışıldığı üzere seçilen 10 rastgele prototipin ortalaması için engelleme oranlarını göstermektedir. Sonuçlar, tüm k değerlerinde MCLV sezgisel yaklaşımının rastgele protip seçimine kıyasla daha az engelleme oranlarına sebep olduğunu göstermekte ve daha faydalı olduğunu doğrulamaktadır. Ayrıca güçlü konum k anonimlik özelliği zayıf konum k-anonimlik özelliğine göre daha katı olduğu için, beklendiği üzere daha yüksek engelleme oranlarıyla sonuçlandığı görülmektedir. Tüm

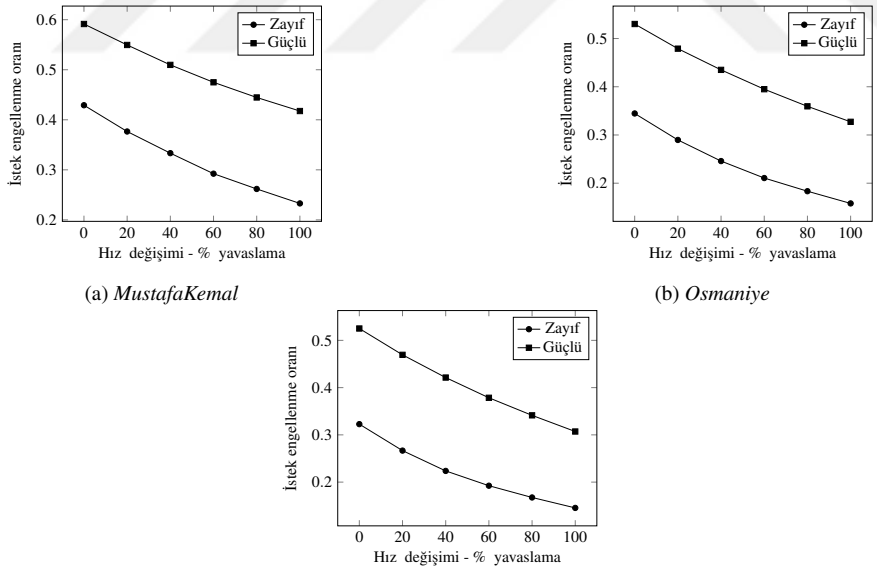


(a) *MustafaKemal* (b) *Osmaniye*

(c) *Ankara*

Şekil 3.7: Zayıf ve Güçlü konum k-anonimliği kavramları için değişen k değerlerine göre konum güncelleme isteği engelleme oranları.

verisetlerinde üzerinde k değerindeki artış, daha yüksek konum anonimliği sunduğu için daha yüksek engellenme oranlarıyla sonuçlanmaktadır ve literatürde iyi bilinen mahremiyet/fayda dengesini doğrulamaktadır.



(a) *MustafaKemal* (b) *Osmaniye*

(c) *Ankara*

Şekil 3.8: Zayıf ve Güçlü konum k-anonimliği kavramları için değişen seyahat hızlarına göre konum güncelleme isteği engelleme oranları.

Algoritma 2'deki $\tau - t$ değeri, bir önceki konum paylaşımı ve şuanki konum paylaşımı arasında geçen seyahat süresini (dolayısıyla hareketlilik hızını) belirtir. Bu değer, hareketlilik hızı arttığında azalır ve hareketlilik hızı azaldığında artar. Bu bağlamda Şekil 3.8, kullanıcı hareket hızındaki değişimin engelleme oranına etkisini incelemektedir.

Grafiklerdeki X ekseni seyahat hızındaki değişim yüzdesini göstermektedir. Örneğin % 0 durumu Şekil 3.7'te bildirilen sonuçlara karşılık gelirken, % 100 durumu ise seyahat hızlarının yarıya indiği ve bu nedenle seyahat sürelerinin iki katına çıktığı duruma karşılık gelir. Sonuçlar, tüm verisetleri ve zayıf/güçlü konum k-anonimlik durumlarında daha yavaş hareket hızının daha az engellenme oranıyla sonuçlanacağını göstermektedir. Bu durum, elbette hareket hızı arttıkça $\tau - t$ değerinin azalması ve Algoritma 2'deki satır 13'ün daha çok çalışmasından kaynaklanır.

4. KENTSEL ALAN HAREKETLİLİĞİNDE ORTAK KONUMLANDIRMA SALDIRILARI ALTINDA ANONİM KONUM PAYLAŞIMI

Önceki bölümde önerilen konum anonimleştirme çatısı, kullanıcılar izole olarak değerlendirildiğinde kullanıcıların saldırgan KTS üzerinden konum k-anonimliği (Tanım 9 ve Tanım 10'a göre) koruyan konum güncellemeleri yapmasına imkan tanır. Kısaca KTS, Tanım 11'de verilen arka plan bilgisi ve hız-tabanlı saldırılardan faydalanarak kullanıcıların konum k-anonimliğini ihlal etmeye çalışır ve bu saldırılar, bir önceki bölümde önerilen Algoritma 2 ve onun zayıf konum k-anonimlik için olan varyantı ile engellenir. Fakat saldırgan KTS, çeşitli yöntemlerle arka plan bilgisine yeni bilgiler ekleyerek kullanıcıların konum k-anonimliğini ihlal etmek amacıyla daha farklı saldırılar düzenleyebilir. Bu amaçla tezin bu bölümünde, saldırgan KTS'nin kullanıcı hareketleri arasındaki ilişkilerden türetebileceği başka bir saldırı türü olan ortak konumlandırma saldırılarına odaklanılmaktadır. Buradaki problem, saldırgan KTS'nin kullanıcıların ortak bir konumda (yani birlikte) bulanabileceği yada ortak hareket edebileceği bilgisini sayısallaştırabileceği gerçeğinden dolayı ortaya çıkar. Bu nedenle KTS, konum güncellemesi yapan kullanıcıların belirli şartlar dahilinde diğer kullanıcılarla ortak bir konumda (yani birlikte) bulunabileceğini çıkarabilir. Bu bağlamda, tezin bu bölümünde bu tip ortak konumlandırma saldırıları detaylandırılmakta ve önceki bölümde önerilen çatıya ek bir mahremiyet modeli olarak konum k-anonimlik modelinin bir varyantı olan ortak konum k-anonimlik modeli tanıtılmaktadır. Formal olarak ifade edersek, hedef bir u kullanıcısının konum k-anonimliğini ihlal etmek amacıyla saldırgan KTS, diğer kullanıcıların konum güncellemelerini kullanarak u kullanıcısının konum bilgisine ilişkin çıkarım kanalları yaratır. Bu çıkarım kanalları, belirli şartlar dahilinde u kullanıcısının anlık konumuna ilişkin bilgi taşır ve bölümün devamında detaylandırılacağı üzere u kullanıcısının konum k-anonimliği ihlal edilebilir. Bu bağlamda tezin bu bölümündeki öneri, önceki bölümde önerilen çatıya ek olarak bir TTP (*Trusted Third Party*) katmanı eklenmesi ve TTP'nin zayıf/güçlü konum k-anonimliği koruyan konum güncellemelerini işleyerek ortak konum k-anonimlik özelliğini koruması yönündedir.

Şekil 4.1'de bir kentsel alan örneği gösterilmektedir. Harita, Ankara şehri içinde yer alan Mustafa Kemal mahallesinin şehir ağını gerçekleştirmektedir. Bu örnekte, Bob ve Tom kullanıcılarının mobil cihazları aracılığıyla KTS'ye bağlandığını varsayalım.

Şekil 4.1: Kentsel alanda konum mahremiyetini ihlal etmek için ortak konum bilgisinin nasıl kullanılabileceğine dair bir örnek durum.

KTS'ye güvenilmediğinden dolayı, hem Bob hem de Tom, KTS'den istedikleri konum mahremiyet seviyelerinde yararlanmak amacıyla kaba (perdelenmiş) konumlarını KTS ile paylaşırlar. Şekil 4.1'deki mavi noktalı dairenin, KTS'nin Bob'un nerede olduğu hakkındaki şu andaki bilgisini temsil ettiğini ve Şekil 4.1'deki siyah noktalı dairenin, KTS'nin Tom'un nerede olduğu hakkındaki şu andaki bilgisini temsil ettiğini ve Bob ve Tom'un şu anda perdelenmiş bölgelerinin kesişme noktasında birlikte bulunduğunu varsayalım (küçük yeşil daire). Bu örnekte saldırgan KTS'nin arkaplan bilgisine göre Bob ve Tom'un aynı yerde bulunma olasılığının yüksek olması durumunda, KTS'nin hem Bob'un hem de Tom'un bulunduğu yeri yüksek bir güvenle küçültebileceği, dolayısıyla KTS'nin her iki kullanıcının da gerçek konumunu elde edebileceği görülmektedir.

4.1 Problem Formülasyonu

Problem formülasyonu, (i) mahremiyet modeli ve (iii) saldırı modelinin detaylandırılmasını içerir. Buradaki öneri bir önceki bölümde önerilen çatıya bir ek niteliğinde olduğundan dolayı, hareketlilik modeli olarak Bölüm 3'te önerilen hareketlilik modeli aynı şekilde kullanılmaktadır. Tekrar hatırlatmak gerekirse, gerçek (kesin) kullanıcı yörüngeleri ACN'e göre tanımlanır. ACN'den üretilen kaba bir çizge modeli olan açıklamalı kaba şehir ağı (ACCN), kaba kullanıcı yörüngelerini tanımlamak için kullanılır. ACN'deki konumlar hassas olduğundan, yalnızca ACCNdeki konumlar KTS sağlayıcısı (saldırgan) ile paylaşılır. Ortak konum k-anonimliği kavramına sahip olan mahremiyet modeli, konum k-anonimliğine dayanmaktadır. Mahremiyet modeliyle eşleşmesi için, ACCN'nin her düğüm noktası ACN'nin bir altçizgesini (en az k düğümlü) temsil eder. Saldırı modeli, ACN, ACCN ve ortak konumda bulunma bilgisinin genel bilgi olduğunu varsayar ve saldırganın görevi, ACCN'de bildirilen konumlardan ACN'deki gerçek kullanıcı konumlarına ilişkin çıkarımlar yapmaktır.

4.1.1 Mahremiyet modeli

Bölüm 3'te tanıtılan konum k-anonimlik varyantları (Tanım 9 ve Tanım 10) bireysel kullanıcıların istenen mahremiyet gereksinimlerini diğer kullanıcılardan izole olduğu düşünülerek tanımlar. Ancak KTS sağlayıcı, diğer kullanıcılardan da servis talepleri alır ve kullanıcı grupları için aynı konumda (ortak konumda) bulunma bilgisi oluşturabilir. Bu bilgi elbette rastgele değildir ve örneğin en iyi arkadaşların (yani, birlikte takılma eğiliminde olan arkadaşlar) nerede olduğu konusunda oldukça bilgilendiricidir. Bu gibi durumlarda, her kullanıcı için izole olarak konum k-anonimliği sağlamak, *sözde konum k-anonimliği* ile sonuçlanabilir. Basitçe varsayarsak, örneğin iki yakın arkadaşın (u_1 ve u_2) zayıf/güçlü konum k-anonimliği izole olarak değerlendirildiğinde sağlanmış olsun, öyle ki $|B^{\tau}_{u_1}(\tau)| \geq k_{u_1}$ ve $|B^{\tau}_{u_2}(\tau)| \geq k_{u_2}$. Fakat u_1 ve u_2 arasındaki ilişkinin derecesini bilen KTS sağlayıcısı, örneğin u_1 kullanıcısının konum bilgisiyle ilgili çıkarımda bulunmak amacıyla u_1 kullanıcısının konumuna ilişkin kanısını $|B^{\tau}_{u_1}(\tau) \bigcap B^{\tau}_{u_2}(\tau)| < k_{u_1}$ olacak şekilde daraltarak u_1 kullanıcısının konum k-anonimliğini ihlal edebilir.

Genelliği kaybetmeden, tezin devamında belirli bir $u_1 \in U$ kullanıcısı, hedef kullanıcı (KTS tarafından) olarak ele alınmaktadır. Yukarıda ele alınan ortak konum bilgisi, u_1 kullanıcısının nerede olduğu konusunda bir tür çıkarım kanalı yaratır. Bu sebeple çıkarım kanalları, bu çalışmada ortak konumda bulunma olasığıyla modellenmiştir. Bu olasılık, 0.0 (iki kullanıcı hiç ilişkili değildir) ile 1.0 (iki kullanıcı kesinlikle aynı konumdadır) arasında herhangi bir değerde olabilir.

Tanım 13 (Ortak konumda bulunma olasılığı) Ortak konumda bulunma olasılığı fonksiyonu (belirli bir hedef kullanıcı u_1 için) $CLP_{u_1} : U \rightarrow [0..1]$, u_1 kullanıcısının diğer kullanıcılarla ortak konumda (birlikte) bulunma olasılığını sayısallaştırır. Örneğin $CLP_{u_1}(u_2)$, u_1 kullanıcısının u_2 kullanıcısıyla ortak bir konumu paylaşma olasılığıdır.

Ortak konumda bulunma olasılığı fonksiyonu CLP_{u_1}, elbette zaman içinde değişebilir ve konumdan konuma (örneğin buluşma yerleri) farklılık gösterebilir. Bu bağlamda CLP_{u_1}'in oluşturulma süreci ayrı bir araştırma konusudur ve yörünge veritabanı üzerinden veri madenciliği yöntemleri kullanılarak bu kişiye özel fonksiyonlar bir saldırgan (KTS) tarafından üretilebilir. Fakat bu tez çalışmasında, CLP_{u_1}'in u_1 kullanıcısının oturumu boyunca statik olduğu varsayılmıştır. Yapılan bu varsayım, CLP_{u_1}'in genellikle uzun bir süre boyunca değişmeyen önceki kanılara (örneğin tarihi konum güncellemelerinden elde edilen) bağlı olmasından dolayı yanıltıcı değildir ve ortak konumlandırma saldırılarının daha basitçe modellenmesine izin verir.

Bir $u_2 \in U$ kullanıcısı tarafından yakın zamanda yapılan herhangi bir (kesin veya kaba) konum paylaşımı, $CLP_{u_1}(u_2) > 0$ olması koşuluyla u_1 kullanıcısının nerede olabileceğiyle ilgili bir çıkarım kanalı yaratır. Öte yandan, u_2 kullanıcısı tarafından

yapılan son konum paylaşımı uzun zaman önceyse, $CLP_{u_1}(u_2) > 0$ olsa bile u_2 kullanıcısının konum paylaşımı bir çıkarım kanalı olarak değerlendirilmez. Bu sebeple çıkarım kanalları, aşağıda tanımlandığı üzere zamansal bir boyuta da sahiptir.

Tanım 14 (Çıkarım kanalı) Şu anki zaman τ ve geçerlilik süresi Δt verildiğinde, bir u_2 kullanıcısı tarafından yapılan son konum paylaşımı (t zaman etiketiyle) eğer (i) $CLP_{u_1}(u_2) > 0$, (ii) $(B^{\tau}_{u_1}(\tau) \cap B^{\tau}_{u_2}(\tau)) \neq \emptyset$, ve (iii) $\tau - t \leq \Delta t$ sağlanırsa, bu konum paylaşımı u_1 kullanıcısının şuanki konumuna yönelik bir çıkarım kanalı oluşturur. $IC^{\tau}_{u_1}(u_2)$ ile gösterilen çıkarım kanalı bir ikili özelliktir, yani sağlanır (geçerli) veya sağlanmaz (geçersiz). Burada geçerli bir çıkarım kanalının, mutlaka u_1 kullanıcısının u_2 kullanıcısıyla aynı konumda bulunmasını gerektirmediğine de dikkat etmek gerekir.

Yalnızca geçerli çıkarım kanalları (Tanım 14'e göre), u_1 kullanıcısının nerede olduğu hakkında bilgi iletir. Geçerli çıkarım kanalları, aşağıda tanımlandığı gibi, arıtılmış sonraki kanıyı elde etmek ve önceki kanı $B^{\tau}_{u_1}(\tau)$'yi küçültmek amacıyla kullanılabilir.

Tanım 15 (Sonraki kanı) u_1 kullanıcısının nerede olduğuna dair şu andaki (önceki) kanı $B^{\tau}_{u_1}(\tau)$ ve çıkarım kanalı $IC^{\tau}_{u_1}(u_2)$ göz önüne alındığında, u_1 kullanıcısının şu anda nerede olduğu $PB^{\tau}_{u_1}(\tau)$ ile gösterilen şu andaki sonraki kanı ile tanımlanır.

Şu andaki sonraki kanıyı güncellemek için, $IC^{\tau}_{u_1}(u_2)$ çıkarım kanalının geçerliliğine bağlı olarak iki durum birbirinden ayrılmaktadır. (i) Çıkarım kanalı $IC^{\tau}_{u_1}(u_2)$ geçerlidir. O zaman,

$$PB^{\tau}_{u_1}(\tau) = \begin{cases} B^{\tau}_{u_1}(\tau) \cap B^{\tau}_{u_2}(\tau), & CLP_{u_1}(u_2) \text{ olasılıkla;} \\ B^{\tau}_{u_1}(\tau), & 1 - CLP_{u_1}(u_2) \text{ olasılıkla.} \end{cases} \tag{4.1}$$

(ii) Çıkarım kanalı $IC^{\tau}_{u_1}(u_2)$ geçersizdir. O zaman,

$$PB^{\tau}_{u_1}(\tau) = B^{\tau}_{u_1}(\tau) \tag{4.2}$$

Tanım 15, herhangi bir geçersiz çıkarım kanalının hedef kullanıcının konumu hakkındaki kanıları güncelleyemeyeceği gerçeğini kullanır, yani bu durumda sonraki kanı önceki kanıyla aynıdır. Bununla birlikte geçerli herhangi bir çıkarım kanalı, $CLP_{u_1}(u_2)$ olasılığı ile sonraki kanı $B^{\tau}_{u_1}(\tau) \cap B^{\tau}_{u_2}(\tau)$ efektif olacağı için ihmal edilemez. Öte yandan, $1 - CLP_{u_1}(u_2)$ olasılığı ile geçerli çıkarım kanalı efektif değildir ve bu nedenle sonraki kanı, önceki kanıyla aynıdır. Herhangi bir geçersiz çıkarım kanalının saldırgan KTS'nin hedef kullanıcı u_1'in konumuyla ilgili çıkarımda bulunmasında bir etkisi olamayacağı için, tezin devamında konum belirsizliği hakkındaki kanıları güncellemek için yalnızca geçerli çıkarım kanalları dikkate alınmaktadır.

Hedef kullanıcıya yönelik şu andaki geçerli çıkarım kanalını $IC^{\tau}_{u_1}(u_2)$ olarak ele alalım. Sonraki kanı $PB^{\tau}_{u_1}(\tau)$, u_1 kullanıcısının şu anki konumuna ilişkin için efektif

belirsizlik bölgesidir. $PB^{\tau}_{u_1}(\tau)$ ifadesi iki koşullu olduğundan ve gerçek koşulun hangisi olduğu bilinemediğinden dolayı, bu efektif belirsizlik bölgesi özel bir şekilde sabitlenemez. Fakat, bu efektif belirsizlik bölgesinin aşağıda verilen efektif (beklenen) kardinalitesi özel bir şekilde hesaplanabilir. Kısaca ifade etmek gerekirse, sonraki kanının efektif (beklenen) kardinalitesi ilgili belirsizlik bölgelerinin kardinalitelerinin ağırlıklı ortalaması olmaktadır.

$$|PB^{\tau}_{u_1}(\tau)| = CLP_{u_1}(u_2) * |B^{\tau}_{u_1}(\tau) \cap B^{\tau}_{u_2}(\tau)| + (1 - CLP_{u_1}(u_2)) * |B^{\tau}_{u_1}(\tau)| \tag{4.3}$$

Yukarıdaki formül, Tanım 15'i kullanarak u_1 kullanıcısının anlık konum belirsizliğini (bir u_2 kullanıcısına göre) sayısallaştırır. Fakat, u_1 kullanıcısı bir arkadaş grubuyla birlikte olabilir, yani mutlaka herhangi bir u_2 kullanıcısıyla belirli bir çift olarak birlikte bulunmak zorunda değildir. Bu durum teknik olarak gruptaki her arkadaş için bir dizi çıkarım kanalına karşılık gelir. Örneğin, u_1 kullanıcısı için geçerli çıkarım kanalı kümesinin $\mathbf{IC}^{\tau}_{u_1} = \{IC^{\tau}_{u_1}(u_{f_1}), IC^{\tau}_{u_1}(u_{f_2}), \ldots, IC^{\tau}_{u_1}(u^{\tau}_{f_n})\}$ olduğunu varsayalım. Bu durumda $\mathbf{IC}^{\tau}_{u_1}$, n farklı arkadaş için n farklı çıkarım kanalı içerir. Bu nedenle u_1 kullanıcısı, geçerli çıkarım kanalı kümesi $\mathbf{IC}^{\tau}_{u_1}$ içindeki n farklı arkadaşının herhangi bir alt kümesiyle birlikte bulunabilir. Açıkça görüleceği üzere bu durum, dikkate alınması gereken 2^n farklı grup kombinasyonu yaratır. Bununla birlikte bu kombinasyonların tümü, aşağıda tanımlandığı üzere ortak konumda bulunma ihtimali olan grup kombinasyonu değildir.

Tanım 16 (Ortak konumda bulunma ihtimali olan grup kombinasyonu) Geçerli çıkarım kanalı kümesi $\mathbf{IC}^{\tau}_{u_1} = \{IC^{\tau}_{u_1}(u_{f_1}), IC^{\tau}_{u_1}(u_{f_2}), \ldots, IC^{\tau}_{u_1}(u_{f_n})\}$ ve ilgili kullanıcı kümesi $U^{\mathbf{IC}^{\tau}_{u_1}} = \{u_{f_1}, u_{f_2}, \ldots, u_{f_n}\}$ verildiğinde, bir grup kombinasyonu $CG^{\tau}_{u_1} = \{u_{fx_1}, u_{fx_2}, \ldots, u_{fx_m}\} \subseteq U^{\mathbf{IC}^{\tau}_{u_1}}$ ($m \leq n$ ve $\forall u_{fx_i} \in \{u_{f_1}, u_{f_2}, \ldots, u_{f_n}\}$), eğer $B^{\tau}_{u_1}(\tau) \cap \bigcap_{u \in \{u_{fx_1}, u_{fx_2}, \ldots, u_{fx_m}\}} B^{\tau}_{u}(\tau) \neq \emptyset$ koşulunu sağlıyorsa, bu grup u_1 kullanıcısıyla şu anda ortak konumda bulunma ihtimali olan bir grup kombinasyonudur. Aksi takdirde, bu gruptaki kullanıcıların u_1 kullanıcısıyla şu anda ortak bir konumda bulunması olası değildir.

Tanım 16'da belirtildiği üzere, eğer bir grup kullanıcının konum kanıları kesişmiyorsa elbette o grubun ortak bir konumda bulunma ihtimali yoktur. Aynı şekilde, eğer bir gruptaki kullanıcıların konum kanıları u_1 kullanıcısının konum kanısıyla kesişmiyorsa bu grup u_1 kullanıcısının ortak konumda bulunma ihtimali olan bir grup kombinasyonu olamaz.

Tüm grup kombinasyonları içinden u_1 kullanıcısıyla şu anda ortak konumda bulunma ihtimali olan gruplardan bir tanesi $CG^{\tau}_{u_1} = \{u_{fx_1}, u_{fx_2}, \ldots, u_{fx_m}\}$ olsun ve bunun tümleyeni $CG^{\tau}_{u_1}{}' = U^{\mathbf{IC}^{\tau}_{u_1}} \setminus CG^{\tau}_{u_1}$ olsun. Bu durumda u_1 kullanıcısına yapılan ortak-

konumlandırma saldırısında, u_1 kullanıcısının $CG^{\tau}_{u_1}$ grubunda ve (dolayısıyla $CG^{\tau}_{u_1}{}'$ grubundaki hiçbir kullanıcıyla) birlikte bulunma olasılığıyla ilgilenilmektedir.

Tanım 17 (Bir grubun ortak konumda bulunma olasılığı) Şu anki zaman τ'da u_1 ile ortak konumda bulunma ihtimali olan bir grup $CG^{\tau}_{u_1} = \{u_{fx_1}, u_{fx_2}, \ldots, u_{fx_m}\}$ verildiğinde, $CLP_{u_1}(CG^{\tau}_{u_1})$ ile gösterilen grubun u_1 ile ortak konumda bulunma olasılığı $CLP_{u_1}(CG^{\tau}_{u_1}) = \prod_{u_{fx_i} \in CG^{\tau}_{u_1}} CLP_{u_1}(u_{fx_i}) \prod_{u_j \in CG^{\tau}_{u_1}{}'} (1 - CLP_{u_1}(u_j))$'dır.
İspat: Tanım gereği, u_1 kullanıcısının $CG^{\tau}_{u_1}$ içindeki her kullanıcıyla birlikte bulunma olasılığı vardır ve $CG^{\tau}_{u_1}{}'$ içindeki her kullanıcıyla birlikte bulunma olasılığı yoktur. $u_x \neq u_y$ olduğunda her $CLP_{u_1}(u_x)$, $CLP_{u_1}(u_y)$ çifti bağımsız olacağından dolayı, formül tanımda verildiği gibi olmaktadır.

u_1 kullanıcısı yalnızca tek bir grup ile ortak konumda olabileceğinden (yalnız olmanın da özel bir grup olduğunu varsayarak) ve hangisinde olduğu bilinmediği için, tüm alternatif ortak konumda bulunma olasılığı olan grup kombinasyonlarının dikkate alınması gerekir. Açık bir şekilde, u_1 kullanıcısının ortak konumda bulunma olasılığı olmayan grup kombinasyonları ile aynı konumu paylaşması olası değildir. Bu nedenle, her ortak konumda bulunma olasılığı olan grup kombinasyonu için efektif olasılıklar atanması gerekir. Neyse ki bu, ortak konumda bulunma olasılığı olan grupların olasılıklarının normalleştirilmesiyle yapılabilir ve böylece tüm olasılıkların toplamı 1.0 olur.

Fakat ne yazık ki u_1 kullanıcısının ortak konumda bulunma olasılığı olan tüm grup kombinasyonlarını değerlendirmek üstel bir algoritma ile sonuçlanacaktır ve n büyük olduğunda izlenebilir değildir. Öte yandan, gerçek hayatta çoğu kullanıcı genellikle tekli, ikili veya üçlü gruplar halinde hareket eder. Bu nedenle, gerçekçi olmak ve daha güvenli bir işlemsel tarafta olmak için, u_1 kullanıcısıyla şu anki zamanda ortak konumda bulunma ihtimali olan grup boyutu $|CG^{\tau}_{u_1}|$'in değeri 0, 1 veya 2 gibi küçük tam sayılarla sınırlanabilir. Bu amaçla aşağıdaki tanım, $|CG^{\tau}_{u_1}|$'in küçük değerlerinde sonraki kanıların kardinalitelerini güncellemek için gereken ifadeleri göstermektedir.

Tanım 18 (Ortak konumda bulunma olasılığı olan küçük gruplar için sonraki kanının kardinalitesi) Şu anki geçerli çıkarım kanalı kümesi
$\mathbf{IC}^{\tau}_{u_1} = \{IC^{\tau}_{u_1}(u_{f_1}), IC^{\tau}_{u_1}(u_{f_2}), \ldots, IC^{\tau}_{u_1}(u_{f_n})\}$ verildiğinde, u_1 kullanıcısıyla şu anki zamanda ortak konumda bulunma olasılığı olan grup $|CG^{\tau}_{u_1}|$ boyutunu aşağıdaki küçük değerlerle sınırladığımızı varsayalım. Bu durumda her bir grup boyutuna ilişkin sonraki kanıların kardinalitelerinin minimum değerleri aşağıdaki gibi hesaplanır.

- $|CG^{\tau}_{u_1}| = 0$, yani u_1 kullanıcısı tek başına hareket eder. Bu durum, bozulmamış olan durumdur. Öyle ki

$$|PB^{\tau}_{u_1}(\tau)| = |B^{\tau}_{u_1}(\tau)|. \tag{4.4}$$

- $|CG^{\tau}_{u_1}| \leq 1$, yani u_1 kullanıcısı en fazla çiftler halinde hareket eder. Bu durumda, u_1 kullanıcısı (i) başka bir arkadaş $u_{fi} \in \{u_{f_1}, u_{f_2}, \ldots, u_{fn}\}$ ile birliktedir, yada (ii) tek başına hareket eder. O zaman, her bir u_{f_i} ayrı ayrı ele alınır ve kardinalitenin minimum olduğu en kötü durum seçilir, yani

$$\begin{aligned} |PB^{\tau}_{u_1}(\tau)| = \arg min_{u_{f_i} \in \{u_{f_1}, u_{f_2}, \ldots, u_{fn}\}} & CLP_{u_1}(u_{f_i}) * |B^{\tau}_{u_1}(\tau) \cap B^{\tau}_{u_{f_i}}(\tau)| \\ & + (1 - CLP_{u_1}(u_{f_i})) * |B^{\tau}_{u_1}(\tau)|. \end{aligned} \tag{4.5}$$

- $|CG^{\tau}_{u_1}| \leq 2$, yani u_1 kullanıcısı en fazla üçlüler halinde hareket eder. Bu durumda, u_1 kullanıcısı (i) başka iki arkadaş $\{u_{fi}, u_{fj}\} \subset \{u_{f_1}, u_{f_2}, \ldots, u_{f_n}\}$ ile birliktedir, (ii) başka bir arkadaş $u_{f_i} \in \{u_{f_1}, u_{f_2}, \ldots, u_{f_n}\}$ ile birliktedir, (ii) başka bir arkadaş $u_{f_j} \in \{u_{f_1}, u_{f_2}, \ldots, u_{f_n}\}$ ile birliktedir, yada (iv) tek başına hareket eder. O zaman, her bir $\{u_{f_i}, u_{f_j}\} \in \{u_{f_1}, u_{f_2}, \ldots, u_{f_n}\}$ için bu durumlar ayrı ayrı ele alınır ve kardinalitenin minimum olduğu en kötü durum seçilir, yani

$$\begin{aligned} |PB^{\tau}_{u_1}(\tau)| = \arg min_{\{u_{f_i}, u_{f_j}\} \in \{u_{f_1}, u_{f_2}, \ldots, u_{fn}\}} & CLP_{u_1}(u_{f_i}) * CLP_{u_1}(u_{fj}) \\ & * |B^{\tau}_{u_1}(\tau) \cap B^{\tau}_{u_{f_i}}(\tau) \cap B^{\tau}_{u_{f_j}}(\tau)| + CLP_{u_1}(u_{f_i}) * (1 - CLP_{u_1}(u_{f_j})) \\ & * |B^{\tau}_{u_1}(\tau) \cap B^{\tau}_{u_{f_i}}(\tau)| + (1 - CLP_{u_1}(u_{f_i})) * CLP_{u_1}(u_{f_j}) \\ & * |B^{\tau}_{u_1}(\tau) \cap B^{\tau}_{u_{f_j}}(\tau)| + (1 - CLP_{u_1}(u_{f_i})) * (1 - CLP_{u_1}(u_{f_j})) \\ & * |B^{\tau}_{u_1}(\tau)|. \end{aligned} \tag{4.6}$$

Tanım 18, hedef bir u_1 kullanıcısına yönelik olan ortak konumlandırma saldırısının son aşaması olan sonraki kanıların kardinalitesinin (ilgili grup boyutu $|CG^{\tau}_{u_1}|$ 'ye göre) sayısal ifadelerini göstermektedir. Tanımdan dikkat etmek gerekirse, u_1 kullanıcısının sonraki kanısının kardinalitesi, ortak konumda bulunma olasılığı fonksiyonu ve kanı kesişimlerinin boyutuna bağlıdır. Ortak konumda bulunma olasılığı fonksiyonu CLP_{u_1}'nin değerleri önceden bellidir ve u_1'in yörüngesi boyunca değişmez. Fakat kanı kesişimleri (dolayısıyla $\mathbf{IC}^{\tau}_{u_1}$'in elemanları), u_1 kullanıcısının yörüngesi boyunca değişkenlik gösterebilir.

Tanım 14'e dikkat edersek u_1 kullanıcısı için şu anki çıkarım kanalı $IC^{\tau}_{u_1}(u_2)$, u_1 kullanıcısının şu anki konumuna ilişkin bilgi taşıyan kullanıcıların konumlarından oluşur ve u_1 kullanıcısıyla şu anki zamanda ortak konumda bulunma olasılığı olan grup $|CG^{\tau}_{u_1}|$ boyutuna göre değişkenlik göstermez. Aslında $IC^{\tau}_{u_1}(u_2)$, tanım 18'e dikkat edersek, $|CG^{\tau}_{u_1}| \leq 1$ durumunda sonraki kanı hesabında kullanılacak kullanıcıları içerir. Fakat $|CG^{\tau}_{u_1}| \leq 2$ durumunda çıkarım kanalı yine aynı olur. Burada hesaba katılacak ikili

$(u_i, u_j) \in IC^{\tau}_{u_1}(u_2)$ kullanıcılar, çıkarım kanalı kümesi içinde ikili olarak ortak konumda bulunma olasılığı olan (Tanım 16'a göre) gruplardan oluşur ve Tanım 18'de verilen $|CG^{\tau}_{u_1}| \leq 2$ durumunda u_1 kullanıcısının sonraki kanı hesabına dahil edilebilmeleri için u_1 kullanıcısıyla da ortak konumda bulunma (yani $B^{\tau}_{u_1}(\tau) \cap B^{\tau}_{u_{f_i}}(\tau) \cap B^{\tau}_{u_{f_j}}(\tau) \neq \emptyset$) olasılığı olan bir grup kombinasyonu oluşturmaları gerekmektedir. Bu bilgiler ışığında, hedef bir u_1 kullanıcısının ortak konumlandırma saldırıları altında konum mahremiyeti aşağıda verilen tanımla modellenmektedir.

Tanım 19 (Ortak konum k-anonimliği) ACN $G = (V, E, w)$, u_1 kullanıcısı için anonimlik parametresi k_{u_1} ve ortak konumda bulunma olasılığı fonksiyonu CLP_{u_1}, şu anki zaman τ, diğer kullanıcılara ilişkin önceki kanılar $B^{t}_{u_{f_i}}(t)$ $(t < \tau)$ ve grup boyutu limiti $|CG^{\tau}_{u_1}| \leq c : c \in [1,2]$ verildiğinde, eğer $|PB^{\tau}_{u_1}(\tau)| \geq k_{u_1}$ ise u_1 kullanıcısı şu anki zaman τ'da ortak konum k-anonimdir.

4.1.2 Saldırı modeli

Hedef $u_1 \in U$ kullanıcısı, ACN'e göre tanımlanmış olan gerçek kullanıcı yörüngesini (TT_{u_1}) gizli tutar ve ACCN'e göre tanımlanmış olan kaba kullanıcı yörüngesini (CT_{u_1}) saldırgan olan KTS'yle paylaşır. Saldırgan, paylaşılan CT_{u_1} güncellemelerinden gizli TT_{u_1}'ya ilişkin çıkarımlarda bulunmaya çalışır. Bu amaçla, saldırgan aşağıdaki Tanım 20'de verilen arka plan bilgisini kullanabilir.

Tanım 20 (Arka plan bilgisi) : Saldırganın arka plan bilgisi şunlardan oluşur :

- Açıklamalı şehir ağı $G = (V, E, w)$
- Her bir kullanıcı $u \in U$ için :
  - Anonimlik parametresi k_u
  - Açıklamalı kaba şehir ağı $G'_{k_u} = (V', E', ew', vw')$
- u_1 kullanıcısı için ortak konumda bulunma olasılığı fonksiyonu CLP_{u_1}

ACN (G), tüm kullanıcılar için aynıdır, fakat k_u anonimlik parametresi ve açıklamalı kaba şehir ağı G'_{k_u} kullanıcıya özgüdür. Saldırganın ayrıca, hedef kullanıcı u_1 için tanımlanan CLP_{u_1} ortak konumda bulunma olasılığı fonksiyonunu da bildiği varsayılmıştır. Bu varsayım altında saldırgan KTS, Bölüm 4.1.1'de detaylandırılan ortak konumlandırma saldırılarından faydalanarak hedef kullanıcı u_1'in gerçek kullanıcı yörüngesine ilişkin çıkarımlarda bulunabilir.

Burada saldırganın görevi, hedef u_1 kullanıcısıyla ortak konumda bulunma olasılığı olan diğer kullanıcıların konum güncellemelerinden (dolayısıyla konum kanılarından) dolayı

oluşan çıkarım kanallarını kullanarak $PB^{\tau}_{u_1}(\tau)$'i küçültmek ve böylece $|PB^{\tau}_{u_1}(\tau)| < k_{u_1}$'i elde ederek hedef u_1 kullanıcısının anlık konum k-anonimliğini bozmaktır.

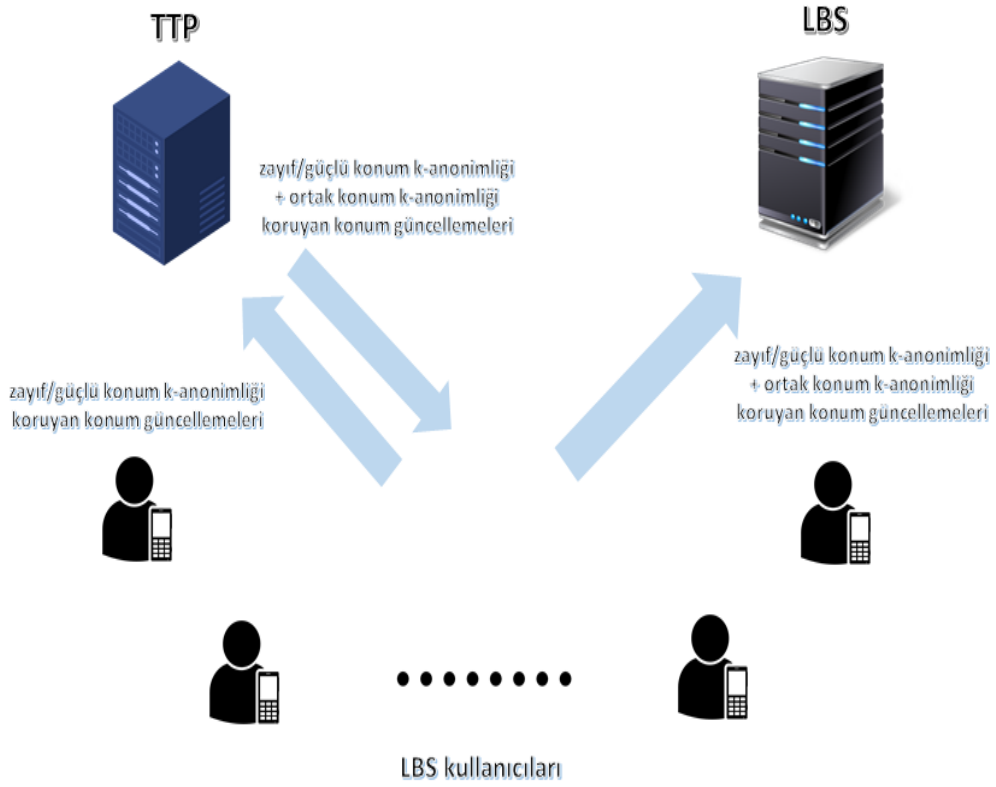
4.2 Konum Anonimleştirme Çatısı

Konum anonimleştirme çatısı, Şekil 4.2'de görüldüğü üzere üçüncü bir güvenilir taraf (TTP) içeren katmanlı bir yapıya sahiptir. Temel olarak, her kullanıcı kendi gerçek kullanıcı yörüngesini gizli tutar ve TTP/KTS ile paylaşmak için ilgili kaba kullanıcı yörüngesini hesaplar. Ancak, zayıf/güçlü konum k-anonimlik ihlallerine karşı ilgili kaba kullanıcı yörüngesinin paylaşılmasının kontrol edilmesi (ve gerekirse engellenmesi) gerekir. Bu amaçla, kullanıcılar Bölüm 3'te zayıf/güçlü konum k-anonimliği korumaya yönelik önerilen algoritmalardan faydalanırlar. Ayrıca, zayıf/güçlü konum k-anonimliğinin sağlanması, ortak konum k-anonimliğinin sağlandığı anlamına gelmez. Kısaca KTS, Bölüm 4.1.1'te anlatılan ortak konumlandırma saldırılarından faydalanarak kullanıcıların konum k-anonimliğini ihlal edebilir. Bu amaçla, adanmış bir TTP, gelen zayıf/güçlü konum k-anonimliği koruyan konum güncelleme isteklerini ortak konumlandırma saldırılarına karşı işlemekten sorumludur ve ortak konum k-anonimliği koruyan konum güncellemelerini KTS'yle paylaşılmak üzere kullanıcalara geri döndürür (ve gerekirse engeller). Bu nedenle, bu bölümde önerilen çatıda TTP yarı güvenilir olarak hizmet eder. Yani kullanıcılar, ortak konum k-anonimliğin sağlanması adına TTP'ye güvenirler. Fakat sadece zayıf/güçlü konum k-anonimliği koruyan konum güncellemelerini (yani gerçek konumları değil) TTP ile paylaşırlar.

Bölüm 3'te zayıf/güçlü konum k-anonimliğinin sağlanması için önerilen algoritmalar kullanıcı tarafında çalışmaktadır. Fakat, bu bölümde önerilen algoritmalar, bir TTP tarafından çalıştırılmaktadır. Ayrıca bu bölümde önerilen algoritmalar, Bölüm 3'te önerilen çatıya bir ek niteliğindedir ve ortak konum k-anonimliği korumak amacıyla ek bir TTP katmanı eklenmektedir.

4.2.1 Ortak konum k-anonimliğin sağlanması

Bu bölümde Tanım 19'a göre ortak konum k-anonimlik özelliğini sağlamaya ilişkin problem ve çözümü sunulmuştur. Önerilen çözüm, hedef bir u_1 kullanıcısından konum güncelleme isteği geldiğinde ortak konum k-anonimlik özelliğinin anlık olarak korunmasına dayanır. Kısaca ifade etmek gerekirse, u_1 kullanıcısından gelen zayıf/güçlü konum k-anonimliği koruyan konum güncelleme istekleri ortak konum yerelleştirme saldırılarına dayalı çıkarımlara karşı üçüncü bir güvenilir taraf (TTP) tarafından kontrol edilir. Ortak konum k-anonimlik ihlali yaratan istekler engellenir. Bölümün devamında



Şekil 4.2: Konum anonimleştirme çatısının katmanlı yapısı.

verilen Problem 3'te bir u_1 kullanıcısının anlık olarak ortak konum k-anonimlik özelliğinin korunulması düşünülmüştür.

Ortak konum yerelleştirme saldırısı (Bölüm 4.1.2'de açıklandığı üzere) kullanıcı etkileşimlerinden kaynaklanır. Bu nedenle, bir u_1 kullanıcısının ortak konum k-anonimliğini korumak için u_1 kullanıcısının konum güncelleme isteğinin KTS'nin diğer kullanıcıların nerede olabileceğine ilişkin kanılarına karşı kontrol edilmesi gerekir. Bu amaçla, güvenilir bir üçüncü taraf (TTP), KTS kullanıcılarından konum güncelleme isteklerini alır ve KTS'nin kullanıcılara ilişkin kanılarını simüle eder. Hedef u_1 kullanıcısının konum güncelleme isteğine karşı oluşan herhangi bir geçerli çıkarım kanalı, Tanım 19'a göre ortak konum k-anonimlik ihlali riskini de beraberinde getirir. Çıkarım kanallarının tanım gereği zamansal bir boyutu da olduğu için, saldırgan (KTS) aşağıda tanımlandığı gibi kullanıcılar hakkındaki kanılarını ekstrapole edebilir.

Tanım 21 (Kanı ekstrapolasyonu) ACN $G = (V, E, w)$ ve KTS'nin bir $u_i \in U$ kullanıcısının en son konumuna ilişkin şu anki kanısı $B^{\tau}_{u_i}(t_i) = P_{u_i} \subseteq V(w.r.t.G'_{k_{u_i}})$ verildiğinde, u_i kullanıcısının şu anki zamana ekstrapole edilmiş kanısı $EPB^{\tau}_{u_i}(\tau) = \{vv \in V : \exists v \in B^{\tau}_{u_i}(t_i) \text{ s.t. } EnKisaYol(v, vv) \leq \tau - t_i\}$'dir. Ayrıca, u_i kullanıcısından τ ve t_i zamanları arasında başka herhangi bir konum güncellemesi gelmediğinden dolayı, $B^{\tau}_{u_i}(\tau) \leftarrow EPB^{\tau}_{u_i}(\tau)$ olur.

Bir u_i kullanıcısı için kanı ekstrapolasyonu, u_i kullanıcısının son konum güncelleme zamanı t_i'den şu anki zamana kadar gidebileceği düğümlerin kümesini ifade eder. Yani u_i kullanıcısının şu anki gerçek kullanıcı yörüngesi $TT_{u_i}(\tau)$, bu düğüm kümesindeki düğümlerden birindedir, öyle ki $TT_{u_i}(\tau) \in EPB^{\tau}_{u_i}(\tau)$.

Problem 3 (Ortak konum k-anonimliğin korunması). Şu anki zaman τ'da bir u_1 kullanıcısından zayıf/güçlü konum k-anonimliği koruyan konum güncelleme isteği geldiğini varsayalım. Tanım 19'a göre ortak konum k-anonimlik özelliğini koruma problemi, $|PB^{\tau}_{u_1}(\tau)| \geq k_{u_1}$ koşulunu sağlamaktır.

Bir u_1 kullanıcısının ortak konum k-anonimliği, u_1 kullanıcısının şu anda nerede olabileceğiyle ilgili olasılıksal bir konum anonimlik modeli sunmaktadır. Ortak konum k-anonimliğin tarihsel olarak sağlanması güçtür. Bunun sebebi, u_1 kullanıcısından şu anki zaman τ'da bir konum güncellemesi gelmediği takdirde bile, KTS'nin çıkarım kanalı geçerlilik süresini ve diğer kullanıcıların konum güncellemelerini kullanarak u_1 kullanıcısına ilişkin şu anki sonraki kanısı $|PB^{\tau}_{u_1}(\tau)|$'i hesaplayabileceği gerçeğinden kaynaklanır. Bu sebeple, bu bölümde ortak konum k-anonimlik özelliğinin anlık olarak sağlanması düşünülmüştür. Yani, hedef u_1 kullanıcısına yönelik oluşan geçerli çıkarım kanallarının sadece u_1 kullanıcısının konum güncellemesi esnasında bilgi taşıdığı varsayılmıştır. Aksi takdirde, TTP'nin diğer bütün kullanıcıların konum güncelleme istekleri esnasında $|PB^{t}_{u_1}(t)|$'i yeniden hesaplaması ve u_1 kullanıcısı için oluşabilecek ortak konum k-anonimlik ihlali durumlarına karşı kontrol etmesi gerekir. Fakat bu durumda, u_1 kullanıcısının ortak konum k-anonimlik özelliğinin sağlanması adına diğer kullanıcıların konum güncelleme istekleri çok sık engellenebilir ve kullanıcıların servisten aldığı fayda ciddi oranda düşebilir. Bu bağlamda, aşağıdaki teoremin kanıtladığı üzere Algoritma 3, bir u_1 kullanıcısı için ortak konum k-anonimlik özelliğini anlık olarak sağlar.

Teorem 3 (Algoritma 3 ortak konum k-anonimlik özelliğini sağlar.) Algoritma 3, bir u_1 kullanıcısından gelen zayıf/güçlü konum k-anonimliğin korunduğu bir konum güncelleme isteği için ortak konum k-anonimlik özelliğini anlık olarak sağlar. Diğer bir deyişle, Tanım 19'a göre ortak konum k-anonimlik ihlali yaratacak herhangi bir bilgi sızdırmaz.

İspat: Şu anki zaman τ'da u_1 kullanıcısından zayıf/güçlü konum k-anonimliği koruyan bir konum güncelleme isteği geldiğinde, KTS'nin diğer kullanıcıların şu anki konumlarına ilişkin şu anki kanıları $B^{\tau}_{u_2}(\tau) = EPB^{\tau}_{u_2}(\tau) : \forall u_2 \in \{U \setminus u_1\}$ olur. Bu kanılar ile birlikte ortak konumda bulunma olasığı fonksiyonu CLP_{u_1} ve çıkarım kanalı geçerlilik süresi Δt, u_1 kullanıcısının şu andaki geçerli çıkarım kanalı kümesi $\mathbf{IC}^{\tau}_{u_1}$'in elemenlarını belirler. Grup boyutu $|CG^{\tau}_{u_1}|$'ye göre (Tanım 18), KTS u_1 kullanıcısı için $\mathbf{IC}^{\tau}_{u_1}$'in her bir elemanını kullanarak bir sonraki kanı $PB^{\tau}_{u_1}(\tau)$ oluşturur. Bu sonraki

kanılar içinden kardinalitesi $|PB^{\tau}_{u_1}(\tau)| \geq k_{u_1}$ olanlar, ortak konum k-anonimlik ihlali yaratmazlar. Fakat $\mathbf{IC}^{\tau}_{u_1}$'in herhangi bir elemanına göre $|PB^{\tau}_{u_1}(\tau)| < k_{u_1}$ olursa, u_1 kullanıcısı için ortak konum k-anonimlik ihlali riski oluşur ve u_1 kullanıcısının konum güncelleme isteği engellenmelidir.

Girdi: ACN $G = (V, E, w)$, kPACN $G_k = (V = \{P_1, P_2, \ldots, P_m\}, E, w)$,
Girdi: prototipli kPACN $G^p_k = (V = \{P^{p_1}_1, P^{p_2}_2, \ldots, P^{p_m}_m\}, E, w)$,
Girdi: ACCN $G'_k = (V', E', ew', vw')$
Girdi: (Eğer mevcutsa) Bir önceki zaman t ve bu zamandaki kanı $B^t_{u_2}(t)$. $\forall u_2 \in U \setminus u_1$ ve $t < \tau$
Girdi: Şu anki zaman τ'da u_1 kullanıcısından gelen ve zayıf/güçlü konum k-anonimlik özelliğini sağlayan konum güncelleme isteği $CT_{u_1}(\tau) = p_j$
Girdi: Grup boyutu limiti $|CG^{\tau}_{u_1}| \leq c$, anonimlik parametresi k_{u_1}
Çıktı: p_j veya **null**.

1: $B^{\tau}_{u_1}(\tau) = P_j$
2: **for** $\forall u_2 \in U \setminus u_1$ **and** $CLP_{u_1}(u_2) > 0$ **do**
3: **if** $B^t_{u_2}(t) = \emptyset$ **or** $\tau - t > \Delta$ **then**
4: **continue**
5: **end if**
6: $B^{\tau}_{u_2}(\tau) \leftarrow EPB^{\tau}_{u_2}(\tau)$
7: **if** $(B^{\tau}_{u_1}(\tau) \cap B^{\tau}_{u_2}(\tau)) \neq \emptyset$ **then**
8: u_2'yi $\mathbf{IC}^{\tau}_{u_1}$'e ekle
9: **end if**
10: **end for**
11: Tanım 18 ve grup boyutu limiti c'ye göre $|PB^{\tau}_{u_1}(\tau)|$'i hesapla
12: **if** $|PB^{\tau}_{u_1}(\tau)| < k_{u_1}$ **then**
13: **return null**, (Konum paylaşma isteği engellenir)
14: **else**
15: **return** p_j (Kullanıcı bu konumu KTS ile paylaşır)
16: **end if**

Algoritma 3: Güvenilir üçüncü taraf (TTP), ortak konumlandırma saldırılarına karşı u_1 kullanıcısının her konum güncellemesinde KTS'nin kanılarını simüle ederek ortak konum k-anonimlik özelliğini sağlar.

TTP, Algoritma 3'te KTS'nin diğer kullanıcıların nerede olduğuna ilişkin geçmiş ve şu anki kanılarını simüle eder. Bir u_1 kullanıcısından şu anki zaman τ'da bir konum güncelleme isteği alındığında, KTS'nin u_1 kullanıcısıyla şu anda ortak konumda bulunma olasılığı olan (yani $CLP_{u_1}(u_2) > 0 : \forall u_2 \in \{U \setminus u_1\}$) diğer kullanıcılara ilişkin şu anki kanısını, KTS'nin bu kullanıcıların konumlarına ilişkin en son kanılarını ekstrapole ederek günceller, yani $B^{\tau}_{u_2}(\tau) \leftarrow EPB^{\tau}_{u_2}(\tau) : \forall u_2 \in \{U \setminus u_1\}$. Tabiki de herhangi bir $u_2 \in \{U \setminus u_1\}$ kullanıcısı için son kanı $B^t_{u_2}(t) = \emptyset$ ise veya $\tau - t > \Delta t$ ise, kanı ekstrapolasyonuna gerek olmaz. Çünkü bu kullanıcının son kanısının Tanım 14'e göre u_1 kullanıcısının konumuna ilişkin bir çıkarım kanalı oluşturma ihtimali yoktur. TTP, sonrasında $(B^{\tau}_{u_1}(\tau) \cap B^{\tau}_{u_2}(\tau)) \neq \emptyset$ olan bütün u_2 kullanıcılarından geçerli çıkarım kanalı kümesi $\mathbf{IC}^{\tau}_{u_1}$'i oluşturur (satır 6-8) ve Tanım 18'e göre u_1 kullanıcısının

kardinalitesi minimum olan sonraki kanısını grup boyutu limiti $|CG^{\tau}_{u_1}|$'ye göre hesaplar (satır 10). Tanım 18'den tekrar hatırlamak gerekirse, $|CG^{\tau}_{u_1}| \leq 1$ durumunda her bir $u_2 \in \mathbf{IC}^{\tau}_{u_1}$ için farklı bir sonraki kanı $|PB^{\tau}_{u_1}(\tau)|$ hesaplanır ve $|CG^{\tau}_{u_1}| \leq 2$ durumunda ise her bir $\{u_2, u_3\} \in \mathbf{IC}^{\tau}_{u_1} : (u_2 \neq u_3)$ için farklı bir sonraki kanı $|PB^{\tau}_{u_1}(\tau)|$ hesaplanır. Sonraki kanıların kardinalitesi minimum olanı, u_1 kullanıcısının sonraki kanısı olarak atanır. Elbette diğer sonraki kanıların kardinalitesi daha büyük olacağından dolayı, ortak konum k-anonimlik ihlali için sadece kardinalitesi minimum olan sonraki kanının kontrolü yeterli olacaktır. Eğer sonraki kanının kardinalitesi $|PB^{\tau}_{u_1}(\tau)| < k_{u_1}$ ise, TTP u_1 kullanıcısının konum güncelleme isteğini engeller (yani **null** döndürür). Aksi takdirde (yani p_j döndürürse), u_1 kullanıcısı bunu güvenle KTS ile paylaşabilir.

Dikkat etmek gerekirse, TTP Algoritma 3'te ortak konum k-anonimliği koruyan konum güncelleme isteğini KTS'ye kendisi göndermez. Bunun sebebi, Bölüm 3'te zayıf/güçlü konum k-anonimliği korumak için önerilen algoritmalarda, kullanıcıların KTS'nin kanılarını simüle etmesinden kaynaklanır. Yani kullanıcının, zayıf/güçlü konum k-anonimliğini koruyabilmesi için KTS'ye iletilen ve konum mahremiyetini koruyan konum güncelleme isteğinden haberdar olması ve KTS'nin kendi konumuyla ilgili kanısını güncelleyebilmesi gerekir. Aksi takdirde kullanıcının zayıf/güçlü konum k-anonimliği ve dolayısıyla ortak konum k-anonimliği sağlanamaz. Eğer TTP ortak konum k-anonimliği koruyan konum güncelleme isteğini KTS'ye kendisi iletirse, kullanıcının zayıf/güçlü konum k-anonimliğini koruyabilmesi için, TTP'nin kullanıcıyı KTS'ye iletilen konum güncellemesi hakkında bilgilendirmesi ve kullanıcının KTS'nin kendi konumu hakkindaki kanısını güncellemesi gerekir.

Hesaplama karmaşıklığı ve iyileştirme TTP'nin yeterli $O(|V|^2)$ alanına sahip olduğunu varsayarsak, bütün düğüm çiftleri arasındaki en kısa yolları önceden hesaplayıp saklayabilir (örneğin Floyd-Warshall algoritmasını kullanarak) ve çevrimiçi sorgu yanıtlama aşamasında en kısa yol uzunluklarını kullanabilir. Bu nedenle, çevrimiçi aşamada herhangi bir düğüm çifti arasındaki en kısa yol mesafeleri $O(1)$ zamanda cevaplanabilir.

Algoritma 3'teki for döngüsünün (satır 2-10) çalışması, sabit bir $m < n$ değeriyle sınırlıdır. Bunun sebebi, genelde günlük hayatımızda arkadaşlarımızla ve yakınlarımızla ortak konumda bulunabileceğimiz ve bunun toplam kullanıcı sayısı $|U| = n$'dan bağımsız bir m değeriyle sınırlı olması (öyle ki $CLP_{u_1}(u_i) > 0 : u_i \in \{u_1, \cdots, u_m\}$) ve dolayısıyla şu andaki geçerli çıkarım kanalının boyutunun da $|\mathbf{IC}^{\tau}_{u_1}| \leq m$ ile sınırlı olacağındandır. 4. satırdaki $EPB^{\tau}_{u_2}(\tau)$, her bir altçizgenin $O(|V|)$ boyutunda olabileceği altçizgeden altçizgeye hesaplamalar gerektirdiği için $O(|V|^2)$ zamanda hesaplanır. Bu sebeple, for döngüsü (satır 2-10) toplamda $O(|V|^2)$ zamanda çalışmaktadır.

Satır 11'deki sonraki kanının kardinalitesinin hesaplamasının en karmaşık durumu olan $|CG^{\tau}_{u_1}| \leq 2$ durumunu ele alalım. $|PB^{\tau}_{u_1}(\tau)|$'in hesaplamasının zaman karmaşıklığı $O(m^2)$ ile sınırlı olacaktır. Çünkü her bir $\{u_{f_i}, u_{f_j}\} \in \mathbf{IC}^{\tau}_{u_1}$ $(u_{f_i} \neq u_{f_j})$ çifti için sonraki kanının kardinalitesinin hesaplanması gereklidir. Bu $O(m^2)$ zaman karmaşıklığı da, toplam kullanıcı sayısı $|U| = n$'den bağımsız olacağı için sabittir. Dolayısıyla, Algoritma 3'ün çalışması toplamda $O(|V|^2)$ zaman karmaşıklığına sahiptir. Ancak, basit bir çizge büyütme tekniği kullanarak ve ardından Dijkstra'nın tek-kaynaklı en kısa yol algoritmasını kullanarak $O(|V|^2)$ zaman karmaşıklığı $O(|V|log|V|)$'a düşürülebilir.

$EPB^{\tau}_{u_2}(\tau)$'ın çalışma zamanını $O(|V|^2)$'dan $O(|V|log|V|)$'a düşürebilmek için, ACN şu şekilde genişletilir. ACN'e tek bir düğüm noktası ekleyerek ve önceki $B^{\tau}_{u_2}(t)$ kanısındaki tüm düğüm noktalarına bu düğüm noktasından sıfır ağırlıklı yönlendirilmiş kenarlar ekleyerek genişletilir. Ardından, yeni eklenen düğüm noktası olarak seçilen kaynak düğüm noktası ile Dijkstra'nın algoritması kullanılır. Ayrıca, erken sonlandırma için $\tau - t$'dan büyük olmayan en kısa yollara sahip düğümleri hesaplamak yeterlidir. Sonuçta ortaya çıkan küme $EPB^{\tau}_{u_2}(\tau)$ olacaktır. Dijkstra algoritmasının karmaşıklığı, Fibonacci heap kullanıldığında $O(|V|\log|V| + |E|)$'dir ve yoğun çizgelerde bu $O(|V|\log|V| + |E| = |V|^2))$ olur. Neyse ki, ACN her bir düğüm noktasının en fazla birkaç düğüm noktasıyla yerel bağlantılara sahip olduğu seyrek bir çizgedir. Dolayısıyla, $O(|V|\log|V| + |E|)$ karmaşıklığı gerçekten de $O(|V|\log|V|)$'a düşer (küçük bir k sabiti için $|E| < k|V|$ olduğundan dolayı). Aslında, uzamsal kısıtlamalar nedeniyle herhangi bir yol kavşağından en fazla birkaç tane kenar çıkmaktadır.

Yukarıda açıklanan çizge genişletme yöntemi, altçizgeden altçizgeye mesafe hesaplama prosedürünün tek düğümden altçizgeye hesaplama prosedürüyle değiştirilmesine izin verir. Toplam $O(|V|\log|V|)$ olan zaman karmaşıklığının güzel yanı, Algoritma 3'ün ACN'nin düğüm sayısıyla yarı doğrusal olarak ölçeklenmesi ve toplam kullanıcı sayısından bağımsız olmasıdır.

4.3 Deneysel Çalışmalar

4.3.1 Deneysel düzenek

Veriseti

Deneyler için Bölüm 3'teki deneyler kapsamında oluşturulmuş olan *MustafaKemal*, *Osmaniye* ve *Ankara* ACN'leri ve ilgili ACCN'ler kullanılmıştır. ACN'lerin özellikleri ve harita düzenleri Çizelge 3.1 ve Şekil 3.5'te gösterilmektedir.

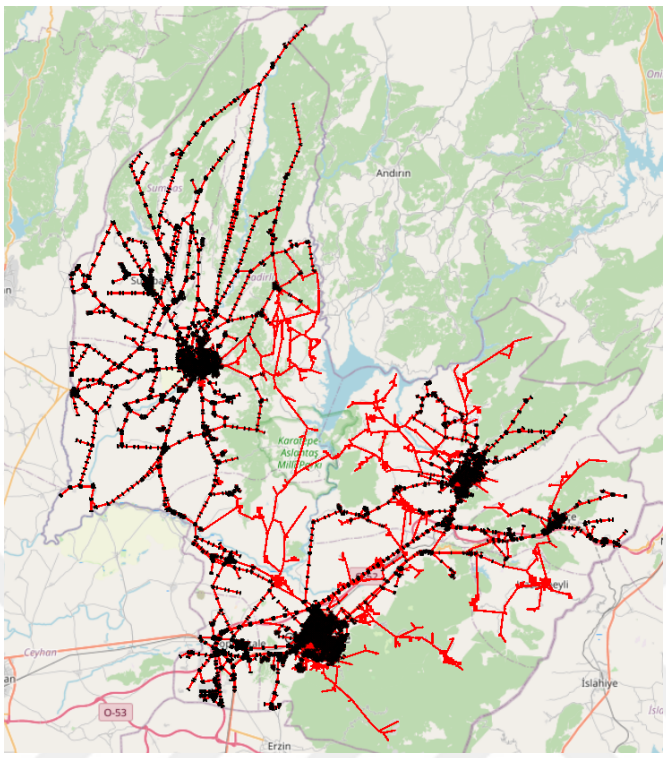
KTS'ye erişim simülasyonu

Kullanıcıların KTS erişimlerini simüle etmek amacıyla, *MustafaKemal* veriseti üzerinde 10 adet kullanıcı için, *Osmaniye* veriseti üzerinde 30 adet kullanıcı için ve *Ankara* veriseti üzerinde 50 adet kullanıcı için sentetik kullanıcı yörüngeleri oluşturuldu. ACN'lerin boyutları aynı olmadığı için, kullanıcı sayıları ilgili ACN'in boyutuna göre kabaca seçilmiştir. Her ACN için kullanıcı kümesi içinden rastgele seçilen bir $u_1 \in U$ kullanıcısı, ortak konum k-anonimlik özelliğinin korunması amacıyla hedef kullanıcı olarak seçilmiş ve diğer bütün $\forall u_2 \in \{U \setminus u_1\}$ kullanıcılar test kullanıcısı olarak kullanılmıştır. Ayrıca kullanıcı kümesindeki tüm kullanıcılar, ilgili ACN üzerinde ortak konumda bulunma ihtimali olan kullanıcılardır. Yani herhangi bir ACN üzerinde $u_1 \in U$ kullanıcısı ve herhangi bir $u_2 \in U \setminus u_1$ kullanıcısı için $CLP_{u_1}(u_2) \neq 0$'dır. Simüle edilmiş yörünge oluşturma süreci, bütün kullanıcılar için aynıdır. Tüm kullanıcılar ortak konumda bulunma ihtimali olan kullanıcılar olduğu için, bu kullanıcıların yolu ilgili ACN üzerinde en az 1 kere kesişmiş olmalıdır. Bu nedenle başlangıçta bütün kullanıcıların konumu, ilgili ACN üzerinde rastgele bir ortak konuma (düğüme) atanır. Sonrasında her bir iterasyonda, kullanıcının gidebileceği rastgele bir düğüm seçilir. Kullanıcının gideceği düğümle şuan bulunduğu düğüm arasındaki kenarın uzunluğunun ew olduğunu varsayalım. Burada, kullanıcıya ew ve $2 * ew$ arasında gerçek bir sayı seyahat süresi olarak atanır. Bu prosedür, kullanıcının bütün uzay-zamansal yörüngesini elde etmek için tekrar edilir. Bu süreçte elde edilen yörüngelerin, yörünge oluşturma süreci ACN'in kenar ağırlıklarına uyduğu için gerçekçi olduğunu belirtmek gerekir. Deneysel çalışmalar kapsamında oluşturulan yörüngelerin uzunlukları *MustafaKemal* için 10000, *Osmaniye* için 200000 ve *Ankara* için 300000 olarak ayarlanmıştır.

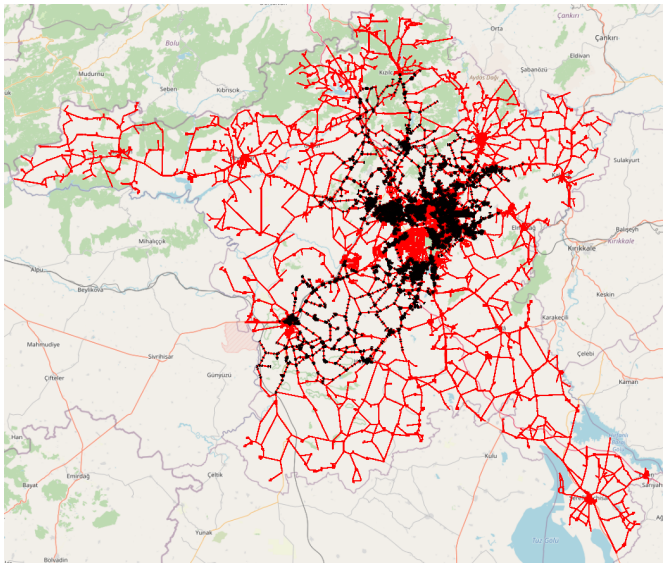
Şekil 4.3'te, deneysel çalışmalar kapsamında oluşturulan yörüngelerin ilgili ACN 'ler üzerinde görselleştirimi gösterilmektedir. Şekilde görülen siyah noktalı çizgiler, her bir ACN için ilgili hedef kullanıcı u_1'e ait yörüngelerdir. Kırmızı çizgiler ise, her bir ACN için ilgili test kullanıcılarının tümünün yörüngelerinin birleştirilmiş halidir. *MustafaKemal* ACN'i küçük bir ACN olduğu için, bu ACN üzerinde hedef kullanıcı ve test kullanıcılarının yörüngeleri tam olarak üstüste binmektedir ve bu sebepten dolayı sadece *Osmaniye* ve *Ankara* ACN'leri için ilgili yörüngelerin görselleştirilmiş hali paylaşılmıştır.

TTP'ye erişim simülasyonu

Ortak konum k-anonimlik özelliğinin korunulması düşünülen hedef u_1 kullanıcısının TTP'ye erişimlerini simüle etmek amacıyla, diğer tüm test kullanıcıları ($\forall u_2 \in \{U \setminus u_1\}$) için Bölüm 3'te önerilen zayıf/güçlü konum k-anonimliği korumaya yönelik algoritmalar ile bu kullanıcıların KTS'ye erişimleri tüm yörüngeleri için ilgili ACCN'ler üzerinde



(a) *Osmaniye* üzerinde oluşturulan gerçek kullanıcı yörüngelerin görselleştirimi



(b) *Ankara* üzerinde oluşturulan gerçek kullanıcı yörüngelerin görselleştirimi

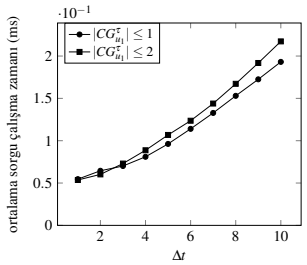
Şekil 4.3: Gerçek kullanıcı yörüngelerin ACN'ler üzerinde gösterimi.

simüle edildi. Bu işlem, bu kullanıcıların farklı mahremiyet seviyeleri için tekrar edildi, öyle ki k_{u_2} değerleri sırayla $k_{u_2} = \{10, 20, 30, 40, 50\} : \forall u_2 \in \{U \setminus u_1\}$ olarak seçildi. Ardından, bu kullanıcıların zayıf/güçlü konum k-anonimliğinin korunduğu konum güncellemeleri ve KTS'nin bu konum güncellemelerine ilişkin kanıları oluşturuldu. Sonrasında bu kanılar, u_1 kullanıcısının TTP'ye erişimlerinde sonraki kanısının kardinalitesini hesaplamak amacıyla kullanıldı.

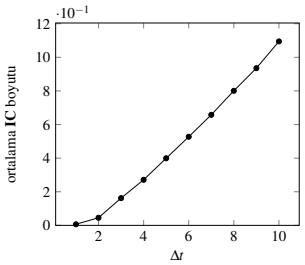
4.3.2 Deneysel sonuçlar

Deneyler, Windows 10 çalıştıran 8 çekirdekli bir dizüstü bilgisayarda (toplam 16 GB RAM ve 2.81 GHz) yapılmıştır. Önerilen algoritma, Java dili kullanılarak uygulanmıştır. Deneylerde, ilk olarak ayarlanabilir tek parametre olan çıkarım kanalı geçerlilik süresinin çalışma zamanına ve çıkarım kanalı boyutuna etkisi ölçülmüştür. Sonrasındaki deneyler, iki farklı senaryoda yapılmıştır. İlk senaryoda, kullanıcıların TTP'den beklediği anonimlik seviyesi ve ACCN'i oluşturmak için kullanılan anonimlik seviyesi (yani k değeri)'nin aynı olduğu varsayılmıştır. Bu senaryoda, çıkarım kanalı parametresinin ve kullanıcıların mahremiyet seviyelerindeki değişimin ortak konum k-anonimlik ihlaline olan etkisi incelenmiştir. Fakat bu senaryoda, ACCN'ler kullanıcıların anonimlik seviyesi k_u'a göre oluşturulduğu için ortak konumda bulunma olasılığı fonksiyonu *CLP*'nin etkisi gözlenebilir değildir. Çünkü herhangi iki kanının kesişimi boş küme değilse (yani $(B^{\tau}_{u_1}(\tau) \cap B^{\tau}_{u_2}(\tau)) \neq \emptyset$), $CLP_{u_1}(u_2) > 0$ olduğu takdirde ortak konum k-anonimlik ihlali ile sonuçlanacaktır. Dahası, kullanıcıların daha katı bir konum mahremiyeti ihtiyacı (örneğin, KTS'den daha iyi gizlenebilmek için) olabilir. Yani, bir u_1 kullanıcısının TTP'den beklediği konum mahremiyeti seviyesi $k_{u_1} = 20$ olabilir ve $k_{u_1} = 50$'ye göre oluşturulmuş bir ACCN kullanabilir. Bu sebeple, deneylerin diğer kısmında bu senaryo gerçekleştirilmiş ve ek olarak *CLP*'nin ortak konum k-anonimlik ihlaline olan etkisi de gözlemlenmiştir. Deneylerin tümü, kullanıcıların zayıf konum k-anonim ve güçlü konum k-anonim olduğu durumlar için ayrı ayrı yapılmış ve ayrıca bu deneyler grup boyutu $|CG^{\tau}_{u_1}| \leq 1$ ve $|CG^{\tau}_{u_1}| \leq 2$ için tekrarlanmıştır. Sonuçlarda zayıf/güçlü konum k-anonimlik durumunun ortak konum k-anonimlik ihlaline ayrıca bir etkisinin olmadığı gözlenmiş ve sadece güçlü konum k-anonimlik durumu için yapılan deney sonuçları raporlanmıştır. Deneylerin tümünde ihlal oranı, ortak konum k-anonimlik ihlali yaratan konum güncelleme sayısının ilgili güçlü konum k-anonimliği koruyan konum güncelleme sayısına oranını göstermektedir. Sonuçlar, mahremiyet/fayda dengesi açısından değerlendirilmiştir.

Çıkarım kanalı geçerlilik süresi Δt'nin etkisi Deneyler kapsamında ilk olarak Δt'nin çalışma zamanına ve şu anki çıkarım kanalı boyutuna etkisi, kullanıcıların güçlü konum k-anonim olduğu durumlarda $|CG^{\tau}_{u_1}| \leq 1$ ve $|CG^{\tau}_{u_1}| \leq 2$ ayarlanarak incelenmiştir.

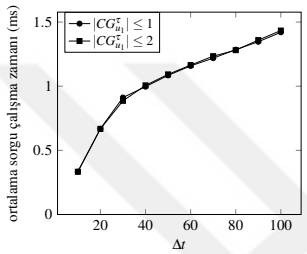


(a) Ortalama TTP sorgusu çalışma zamanı

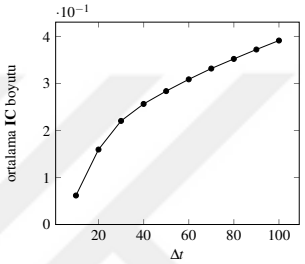


(b) Ortalama geçerli çıkarım kanalı kümesi boyutu

Şekil 4.4: MustafaKemal ACN'i üzerinde çıkarım kanalı geçerlilik süresi Δt'nin çalışma zamanı ve geçerli çıkarım kanalı kümesi boyutuna etkisi. Şekil 4.4a ve Şekil 4.4b'de sırasıyla, Δt'nin değişen değerlerine göre hedef u kullanıcısının tüm yörüngesi üzerinde Algoritma 3'ün ortalama çalışma zamanı ve tespit edilen ortalama geçerli çıkarım kanalı kümesi boyutu gösterilmektedir. $k_u = 50{:}\forall u \in U$.

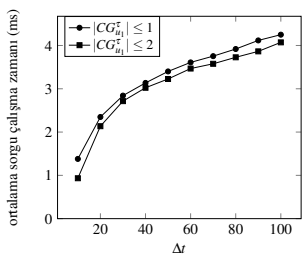


(a) Ortalama TTP sorgusu çalışma zamanı

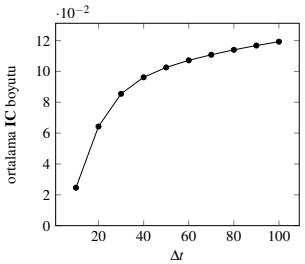


(b) Ortalama geçerli çıkarım kanalı kümesi boyutu

Şekil 4.5: Osmaniye ACN'i üzerinde çıkarım kanalı geçerlilik süresi Δt'nin çalışma zamanı ve geçerli çıkarım kanalı kümesi boyutuna etkisi. Şekil 4.5a ve Şekil 4.5b'de sırasıyla, Δt'nin değişen değerlerine göre hedef u kullanıcısının tüm yörüngesi üzerinde Algoritma 3'ün ortalama çalışma zamanı ve tespit edilen ortalama geçerli çıkarım kanalı kümesi boyutu gösterilmektedir. $k_u = 50{:}\forall u \in U$.



(a) Ortalama TTP sorgusu çalışma zamanı



(b) Ortalama geçerli çıkarım kanalı kümesi boyutu

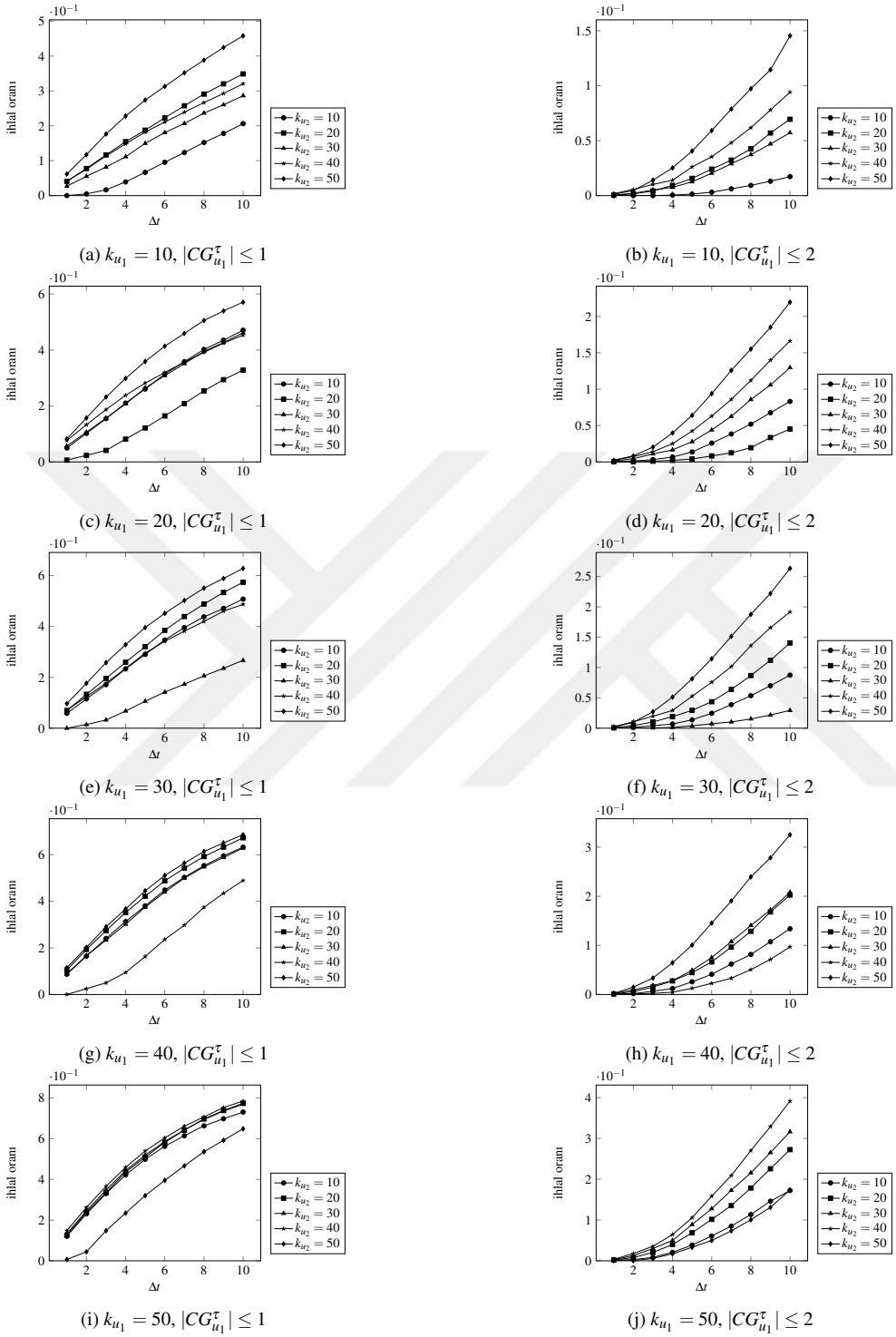
Şekil 4.6: Ankara ACN'i üzerinde çıkarım kanalı geçerlilik süresi Δt'nin çalışma zamanı ve geçerli çıkarım kanalı kümesi boyutuna etkisi. Şekil 4.6a ve Şekil 4.6b'de sırasıyla, Δt'nin değişen değerlerine göre hedef u kullanıcısının tüm yörüngesi üzerinde Algoritma 3'ün ortalama çalışma zamanı ve tespit edilen ortalama geçerli çıkarım kanalı kümesi boyutu gösterilmektedir. $k_u = 50{:}\forall u \in U$.

Şekil 4.4, 4.5 ve 4.6, sırasıyle *MustafaKemal*, *Osmaniye* ve *Ankara* ACN'leri üzerinde Δt'nin ortalama geçerli çıkarım kanalı kümesi boyutu $\mathbf{IC}^{\tau}_{u_1}$'ye ve çalışma zamanına olan etkisini inceler. Bu kısımda kullanıcıların mahremiyet seviyeleri $k_u = 50 : \forall u \in U$ ve $CLP_{u_1}(u_2) = 0.5 : \forall u_2 \in \{U \setminus u_1\}$ olarak sabitlenmiştir. Ayrıca $CLP_{u_1}(u_2)$'in değeri değiştirilerek deneyler yapılmış ve iddia edildiği üzere bu değişimin sonuçlarda kayda

değer bir etkisi görülmemiştir. Şekil 4.4b, 4.5b ve 4.6b'de görüldüğü üzere ortalama geçerli çıkarım kanalı kümesi boyutu $\mathbf{IC}_{u_1}^{\tau}$'deki artış, Δt arttıkça $EPB_{u_2}^{\tau}(\tau)$ daha büyük bir alan kaplayacağı ve Algoritma 3'teki $(B_{u_1}^{\tau}(\tau) \cap B_{u_2}^{\tau}(\tau)) \neq \emptyset$ koşulunun daha çok tutacağı beklentisiyle uyumludur. Çalışma zamanı sonucu (Şekil 4.4a, 4.5a, 4.6a), Δt arttıkça kanı ekstrapolasyonun hesaplanma süresindeki artış nedeniyle Algoritma 3'ün daha yavaş çalışacağını doğrulamakta ve $|CG_{u_1}^{\tau}|$'in farklı değerlerinde beklendiği üzere algoritmanın çalışma süresinde kayda değer bir fark görülmemektedir.

Senaryo 1: Deneylerin bu kısmında, hedef u_1 kullanıcısının TTP'den beklediği anonimlik seviyesi ve ACCN'lerini oluşturmak için kullanılan anonimlik seviyesi k_{u_1} aynıdır. Şekil 4.7, 4.8 ve 4.9'de, sırasıyla *MustafaKemal*, *Osmaniye* ve *Ankara* ACN'leri üzerinde Δt'nin ortak konum k-anonimlik ihlali oranına olan etkisi birkaç farklı ayar üzerinden incelenmektedir. Deneylerde, u_1 kullanıcısının farklı anonimlik seviyeleri için, test kullanıcılarının anonimlik seviyeleri $(k_{u_2} : \forall u_2 \in U \setminus u_1)$ değiştirilerek deneyler yapılmış ve sonuçları raporlanmıştır. Ortak konumda bulunma olasılığı fonksiyonu *CLP* ise tüm kullanıcılar için $CLP_{u_1}(u_2) = 0.5 : \forall u_2 \in \{U \setminus u_1\}$ olarak atanmıştır. Ayrıca, bu fonksiyonun değeri değiştirilerek deneyler tekrar edilmiş ve beklenildiği üzere sonuçlar üzerinde gözlenebilir bir etkisi olmadığı görülmüştür. Şekil 4.7, 4.8 ve 4.9'deki tüm durumlarda görüldüğü üzere, Δt parametresindeki artışın ortak konum konum k-anonimlik ihlali oranını ciddi ölçüde arttırdığı ve bu artışın monoton olduğu gözlemlenmektedir. Dolayısıyla Δt değeri arttıkça, u_1 kullanıcısının ortak konum k-anonimliğini sağlayabilmek adına konum güncelleme istekleri daha sık engellenir ve u_1 kullanıcısının KTS'den aldığı fayda düşmektedir.

Deneysel sonuçları tümü, mahremiyet fayda dengesini doğrulamaktadır. Yani hedef kullanıcı u_1'in mahremiyet seviyesinin artması, ortak konum k-anonimlik özelliğinin daha fazla ihlal edilmesi nedeniyle konum güncelleme isteklerinin daha sık engellenmesine neden olur. Ayrıca sonuçlarda görüldüğü üzere, $|G| = 2$ durumu $|CG_{u_1}^{\tau}| \leq 1$ durumuna göre daha az ortak konum k-anonimlik ihlali ile sonuçlanmaktadır. Bunun sebebi, elbette $|CG_{u_1}^{\tau}| \leq 2$ durumunda hedef u_1 kullanıcısının ortak konum k-anonimlik özelliğinin ihlal edilebilmesi için birden fazla test kullanıcısının ve u_1 kullanıcısının konum kanılarının kesişmesi gerekmesinden dolayıdır.



(a) $k_{u_1} = 10$, $|CG^{\tau}_{u_1}| \leq 1$

(b) $k_{u_1} = 10$, $|CG^{\tau}_{u_1}| \leq 2$

(c) $k_{u_1} = 20$, $|CG^{\tau}_{u_1}| \leq 1$

(d) $k_{u_1} = 20$, $|CG^{\tau}_{u_1}| \leq 2$

(e) $k_{u_1} = 30$, $|CG^{\tau}_{u_1}| \leq 1$

(f) $k_{u_1} = 30$, $|CG^{\tau}_{u_1}| \leq 2$

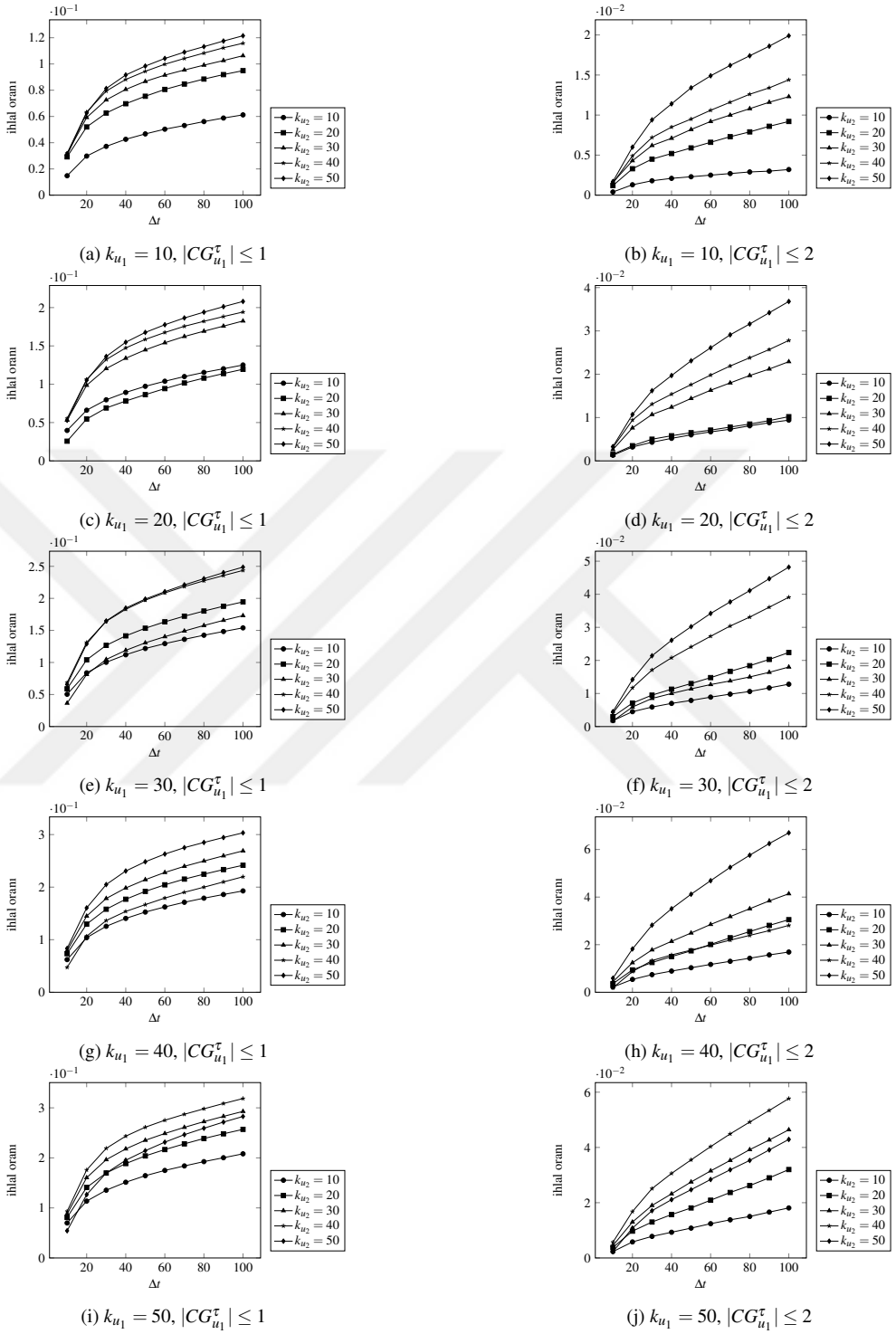
(g) $k_{u_1} = 40$, $|CG^{\tau}_{u_1}| \leq 1$

(h) $k_{u_1} = 40$, $|CG^{\tau}_{u_1}| \leq 2$

(i) $k_{u_1} = 50$, $|CG^{\tau}_{u_1}| \leq 1$

(j) $k_{u_1} = 50$, $|CG^{\tau}_{u_1}| \leq 2$

Şekil 4.7: MustafaKemal ACN'i üzerinde Δt'nın ortak konum k-anonimlik ihlaline etkisi. Şekil (4.7a-4.7c-4.7e-4.7g-4.7i) ve Şekil (4.7b-4.7d-4.7f-4.7h-4.7j) sırasıyla, grup boyutu limiti $|CG^{\tau}_{u_1}| \leq 1$ ve $|CG^{\tau}_{u_1}| \leq 2$ ayarlanarak kullanıcıların farklı mahremiyet seviyeleri için yapılan deney sonuçlarını göstermektedir.



(a) $k_{u_1} = 10$, $|CG^{\tau}_{u_1}| \leq 1$

(b) $k_{u_1} = 10$, $|CG^{\tau}_{u_1}| \leq 2$

(c) $k_{u_1} = 20$, $|CG^{\tau}_{u_1}| \leq 1$

(d) $k_{u_1} = 20$, $|CG^{\tau}_{u_1}| \leq 2$

(e) $k_{u_1} = 30$, $|CG^{\tau}_{u_1}| \leq 1$

(f) $k_{u_1} = 30$, $|CG^{\tau}_{u_1}| \leq 2$

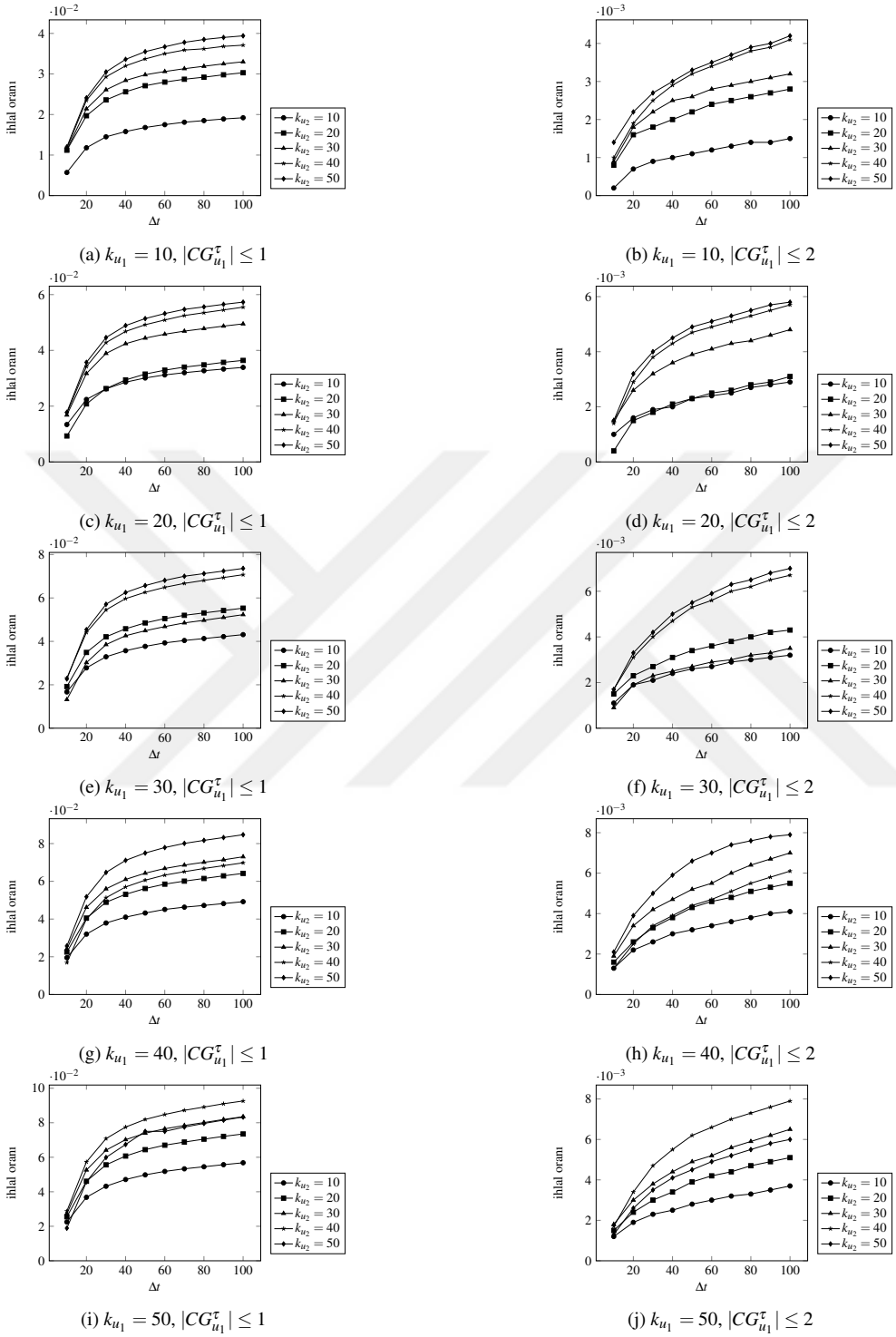
(g) $k_{u_1} = 40$, $|CG^{\tau}_{u_1}| \leq 1$

(h) $k_{u_1} = 40$, $|CG^{\tau}_{u_1}| \leq 2$

(i) $k_{u_1} = 50$, $|CG^{\tau}_{u_1}| \leq 1$

(j) $k_{u_1} = 50$, $|CG^{\tau}_{u_1}| \leq 2$

Şekil 4.8: Osmaniye ACN'i üzerinde Δt'nın ortak konum k-anonimlik ihlaline etkisi. Şekil (4.8a-4.8c-4.8e-4.8g-4.8i) ve Şekil (4.8b-4.8d-4.8f-4.8h-4.8j) sırasıyla, grup boyutu limiti $|CG^{\tau}_{u_1}| \leq 1$ ve $|CG^{\tau}_{u_1}| \leq 2$ ayarlanarak farklı mahremiyet seviyeleri için yapılan deney sonuçlarını göstermektedir.



(a) $k_{u_1} = 10$, $|CG^{\tau}_{u_1}| \leq 1$

(b) $k_{u_1} = 10$, $|CG^{\tau}_{u_1}| \leq 2$

(c) $k_{u_1} = 20$, $|CG^{\tau}_{u_1}| \leq 1$

(d) $k_{u_1} = 20$, $|CG^{\tau}_{u_1}| \leq 2$

(e) $k_{u_1} = 30$, $|CG^{\tau}_{u_1}| \leq 1$

(f) $k_{u_1} = 30$, $|CG^{\tau}_{u_1}| \leq 2$

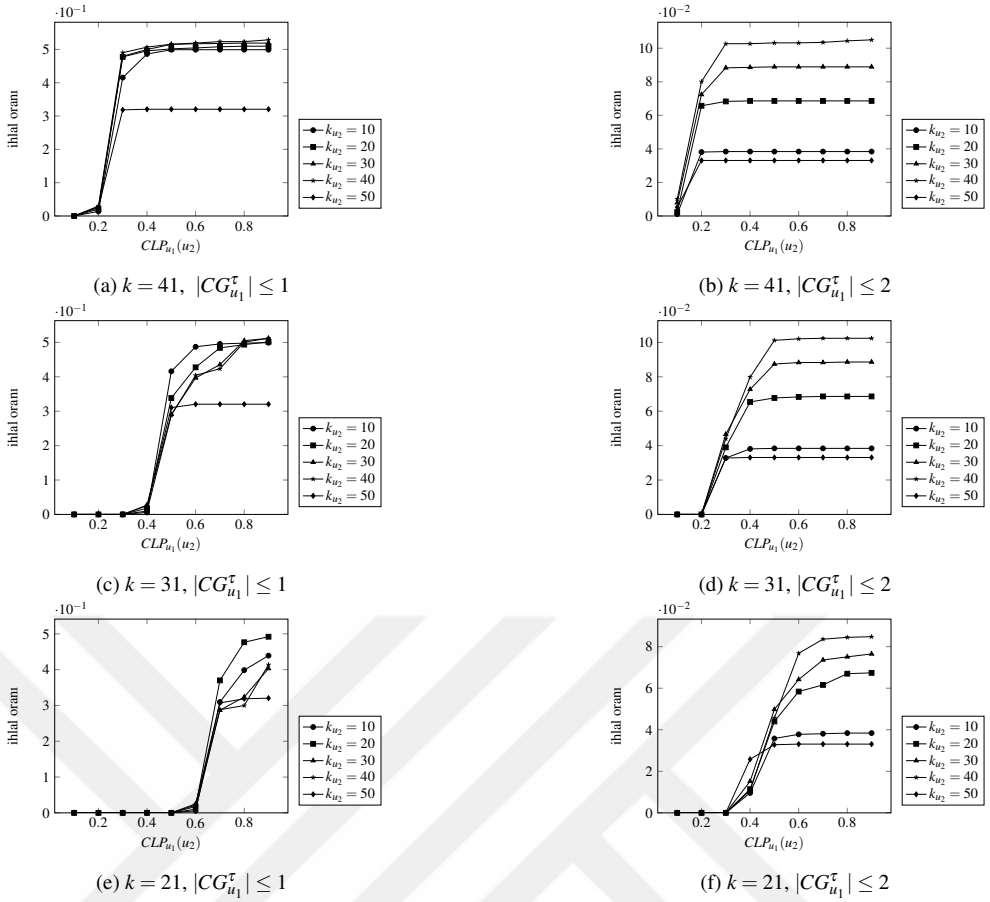
(g) $k_{u_1} = 40$, $|CG^{\tau}_{u_1}| \leq 1$

(h) $k_{u_1} = 40$, $|CG^{\tau}_{u_1}| \leq 2$

(i) $k_{u_1} = 50$, $|CG^{\tau}_{u_1}| \leq 1$

(j) $k_{u_1} = 50$, $|CG^{\tau}_{u_1}| \leq 2$

Şekil 4.9: Ankara ACN'i üzerinde Δt'nın ortak konum k-anonimlik ihlaline etkisi. Şekil (4.9a-4.9c-4.9e-4.9g-4.9i) ve Şekil (4.9b-4.9d-4.9f-4.9h-4.9j) sırasıyla, grup boyutu limiti $|CG^{\tau}_{u_1}| \leq 1$ ve $|CG^{\tau}_{u_1}| \leq 2$ ayarlanarak farklı mahremiyet seviyeleri için yapılan deney sonuçlarını göstermektedir.



(a) $k = 41$, $|CG^{\tau}_{u_1}| \leq 1$

(b) $k = 41$, $|CG^{\tau}_{u_1}| \leq 2$

(c) $k = 31$, $|CG^{\tau}_{u_1}| \leq 1$

(d) $k = 31$, $|CG^{\tau}_{u_1}| \leq 2$

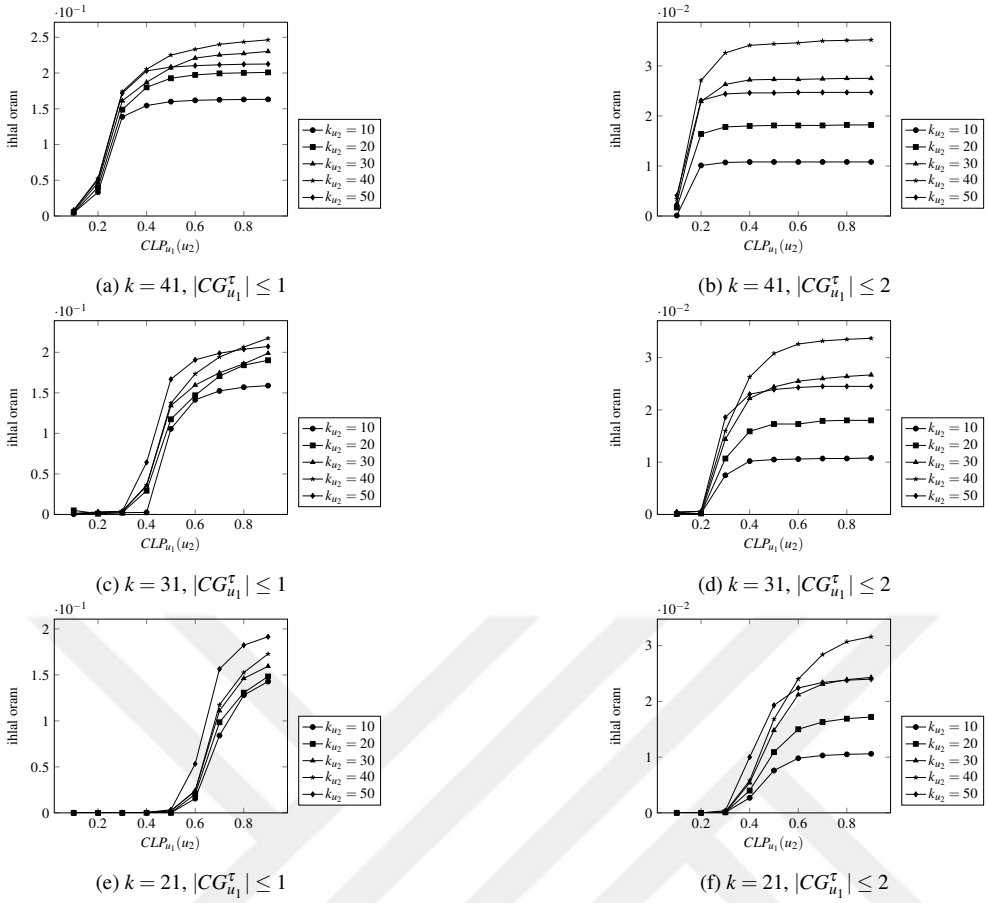
(e) $k = 21$, $|CG^{\tau}_{u_1}| \leq 1$

(f) $k = 21$, $|CG^{\tau}_{u_1}| \leq 2$

Şekil 4.10: MustafaKemal ACN'i üzerinde *CLP*'nin ortak konum k-anonimlik ihlaline etkisi. Şekil (4.10a-4.10c-4.10e) ve Şekil (4.10b-4.10d-4.10f) sırasıyla, grup boyutu limiti $|CG^{\tau}_{u_1}| \leq 1$ ve $|CG^{\tau}_{u_1}| \leq 2$ ayarlanarak kullanıcıların farklı mahremiyet seviyeleri için yapılan deney sonuçlarını göstermektedir.

Senaryo 2 Kullanıcılar KTS'den daha iyi gizlenebilmek adına TTP'den daha katı bir konum mahremiyeti isteyebilir. Örneğin hedef u_1 kullanıcısı, hareketlilik modelinde $k_{u_1} = 50$'ye göre oluşturulmuş olan bir ACCN'i kullanabilir ve (TTP'den) beklediği anonimlik seviyesi $k_{u_1} = 30$ olabilir. Bu nedenle ve $CLP_{u_1}(u_2)$'nin ortak konum k-anonimlik ihlaline olan etkisini gözlemleyebilmek amacıyla, deneylerin bu kısımında *MustafaKemal*, *Osmaniye* ve *Ankara* ACN'leri için hedef kullanıcı u_1'in hareketlilik modeli olarak $k_{u_1} = 50$ değerine göre oluşturulan ACCN kullanılmış (yani her bölütünde en az 50 düğüm içeren bir ACCN) ve sırasıyla (TTP'den) beklediği anonimlik seviyesi $k_{u_1} = \{41, 31, 21\}$ seçilerek deneyler yapılmıştır.

Deneyler birkaç farklı ayar üzerinden yapılıp $CLP_{u_1}(u_2)$'nin ortak konum k-anonimlik ihlaline olan etkisi araştırılmıştır. Bu kısımdaki deneylerde Δt'nin değeri, *MustafaKemal* ACN'i için $\Delta t = 5$, *Osmaniye* ve *Ankara* ACN'leri için ise $\Delta t = 50$ olarak sabitlenmiştir. Deneyler, bir önceki senaryoda olduğu gibi test kullanıcılarının farklı anonimlik seviyeleri için tekrar edilmiş ve sonuçları raporlanmıştır. Şekil 4.10, 4.11, 4.12'de raporlanan deney sonuçlarında, $CLP_{u_1}(u_2)$'in ortak konum k-anonimlik ihlali oranına



(a) $k = 41$, $|CG^{\tau}_{u_1}| \leq 1$

(b) $k = 41$, $|CG^{\tau}_{u_1}| \leq 2$

(c) $k = 31$, $|CG^{\tau}_{u_1}| \leq 1$

(d) $k = 31$, $|CG^{\tau}_{u_1}| \leq 2$

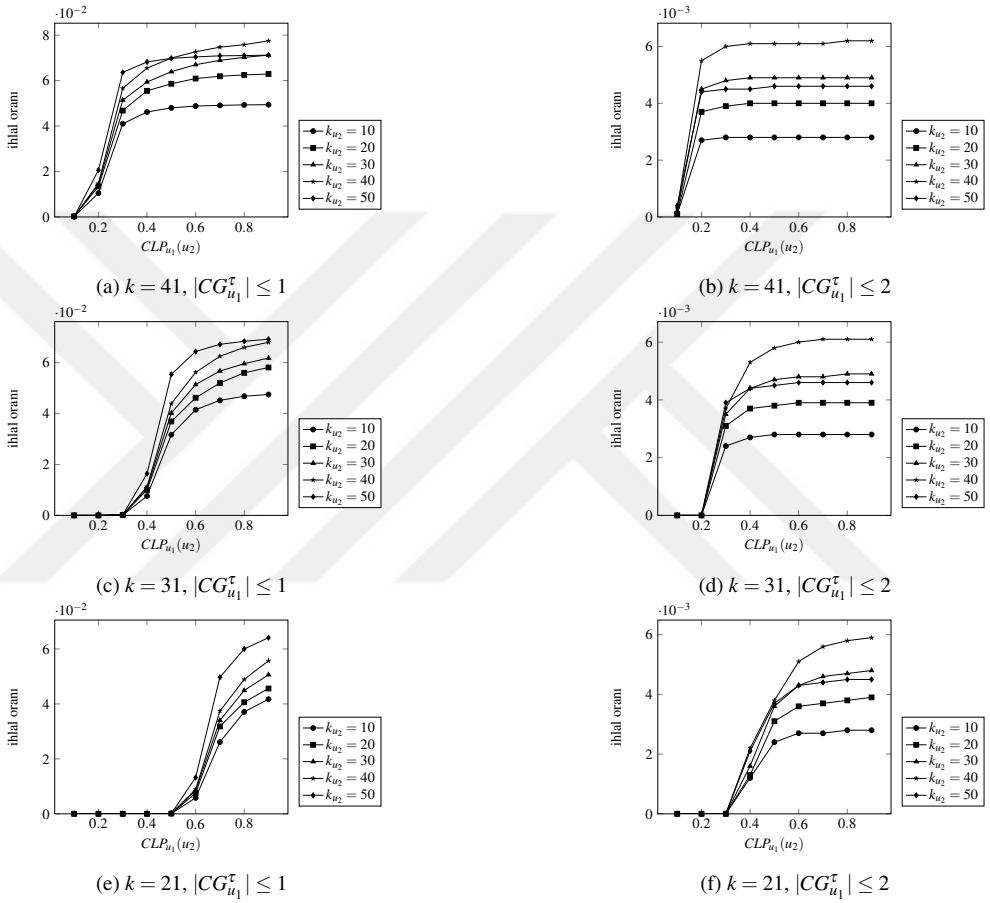
(e) $k = 21$, $|CG^{\tau}_{u_1}| \leq 1$

(f) $k = 21$, $|CG^{\tau}_{u_1}| \leq 2$

Şekil 4.11: Osmaniye ACN'i üzerinde *CLP*'nin ortak konum k-anonimlik ihlaline etkisi. Şekil (4.11a-4.11c-4.11e) ve Şekil (4.11b-4.11d-4.11f) sırasıyla, grup boyutu limiti $|CG^{\tau}_{u_1}| \leq 1$ ve $|CG^{\tau}_{u_1}| \leq 2$ ayarlanarak kullanıcıların farklı mahremiyet seviyeleri için yapılan deney sonuçlarını göstermektedir.

olan etkisinin sadece belirli aralıklarda monoton olduğu görülmektedir. Bu elbette, tüm olası $|PB^{\tau}_{u_1}(\tau)| < k_{u_1}$ koşullarının bütün $u_2 \in U \setminus u_1$ kullanıcıları için $CLP_{u_1}(u_2)$'nin belirli bir eşik değerinden sonra sağlanmaya başlayacağını ve belirli bir eşik değerinden sonra sağlanmış olacağı beklentisini doğrulamaktadır. Ayrıca sonuçlarda görüldüğü üzere, hedef kullanıcı u_1'in TTP'den beklediği anonimlik düzeyi k_{u_1} azaldıkça, $CLP_{u_1}(u_2)$'in ortak konum k-anonimlik ihlaline etkisinin monoton olduğu aralıkların eşik değerleri artmaktadır. Bu elbette, k_{u_1} azaldıkça $|PB^{\tau}_{u_1}(\tau)| < k_{u_1}$ koşulunu sağlamak için daha yüksek $CLP_{u_1}(u_2)$ değerleri gerekeceği beklentisiyle uyumludur. Ayrıca, bu senaryoda yapılan deneyler de $|CG^{\tau}_{u_1}| \leq 2$ durumunun $|CG^{\tau}_{u_1}| \leq 1$ durumuna göre daha az ortak konum k-anonimlik ihlali ile sonuçlandığı doğrulanmaktadır.

Bütün deneysel sonuçlarda, ACN'in boyutu büyüdükçe hedef kullanıcı u_1 için ortak konum k-anonimlik ihlali oranlarının düştüğü gözlenmektedir. Ortak konum k-anonimlik ihlali, hedef kullanıcı u_1 ve test kullanıcılarının konum kanılarının kesişimine bağlı olduğu için, ACN'in boyutu büyüdükçe kanı kesişimlerine daha az rastlanması ve ortak konum k-anonimlik ihlali oranlarının düşmesi elbette şaşırtıcı değildir.



(a) $k = 41$, $|CG^{\tau}_{u_1}| \leq 1$

(b) $k = 41$, $|CG^{\tau}_{u_1}| \leq 2$

(c) $k = 31$, $|CG^{\tau}_{u_1}| \leq 1$

(d) $k = 31$, $|CG^{\tau}_{u_1}| \leq 2$

(e) $k = 21$, $|CG^{\tau}_{u_1}| \leq 1$

(f) $k = 21$, $|CG^{\tau}_{u_1}| \leq 2$

Şekil 4.12: Ankara ACN'i üzerinde *CLP*'nin ortak konum k-anonimlik ihlaline etkisi. Şekil (4.12a-4.12c-4.12e) ve Şekil (4.12b-4.12d-4.12f) sırasıyla, grup boyutu limiti $|CG^{\tau}_{u_1}| \leq 1$ ve $|CG^{\tau}_{u_1}| \leq 2$ ayarlanarak kullanıcıların farklı mahremiyet seviyeleri için yapılan deney sonuçlarını göstermektedir.

5. SONUÇ

Konum tabanlı servisler ve kullanıcı sayılarının artmasıyla beraber, konum mahremiyeti kullanıcılar için önemli bir konu haline gelmiş durumdadır. Bir çok konum tabanlı servis, kullanıcılarına konum güncelleme hizmeti sunmaktadır. Bir konum güncellemesi, anlamlı bir yerdeyken mevcut konumu KTS ile paylaşarak yapılır. Fakat burada akla gelecek ilk soru, KTS'nin güvenilir olup olmadığıdır. Konum bilgisi, kişiler hakkında izinsiz çıkarımlarda bulunmak için kullanılabilen bir bilgi olduğundan dolayı, bu bilginin kötü amaçlı kişilerin eline geçmesi kullanıcıların istemeyeceği bir durumdur. Bu nedenle bu tez çalışmasında, insan hareketliliği uygulamaları kapsamında kentsel alan hareketliliğinde konum mahremiyeti korumalı konum güncellemesi problemi ele alınmıştır.

Tezin ilk bölümde, ilgili probleme yönelik hareketlilik modeli, mahremiyet modeli ve saldırı modeli ayrıntılı olarak verilmiştir. Kentsel alan hareketlilik modeline göre, kullanıcı hareketliliği ağırlıklı yönlü bir çizge olarak modellenen açıklamalı şehir ağı (ACN) üzerinde sınırlandırılmıştır. KTS sağlayıcısının saldırgan olduğu varsayıldığından, kullanıcılar anonim hale getirildikten sonra konumlarını paylaşırlar. Paylaşılan konumlar, ACN'nin prototipli k-üyeli bölütlenmesi ile elde edilen alt çizgelerden oluşur. Mahremiyet modeli, paylaşılan konumların, konum k-anonimliğine dayanır. Saldırı modeli, veri alıcısının kanı modellemesine ve kullanıcıların nerede olduğuna dair kanı güncellemesine dayanır. Kişiselleştirilmiş k değeri, istenen anonimlik seviyesidir.

Çözüm yaklaşımı, bölüt başına bir prototip düğüm noktası seçerek bölütlenmiş ACN'den ACN'nin kaba bir versiyonunu (ACCN) oluşturmayı içerir. Elde edilen ACCN, düğüm noktası ağırlıkları eklenmiş ağırlıklı yönlü bir çizgedir. Burada (i) kompakt bir ACCN'nin nasıl elde edileceği (çevrimdışı aşamada) ve (ii) konum k-anonimliğinin nasıl sağlanacağı (çevrimiçi aşamada) olarak 2 araştırma problemi tanımlanmıştır. İlki (i) k-üyeli bölütleme ve (ii) bölüt başına prototip düğüm noktası seçimi olmak üzere iki adım içerir. K-üyeli bölütleme, NP-Zor bir problem olduğu için literatürden sık kullanılan bir yöntem bu amaçla kullanılmış ve bunun yerine prototip seçimi adımına odaklanılmıştır. Bu amaçla, ACCN'nin tüm kenar ve düğüm ağırlıklarının toplamı olarak tanımlanan kompaktlığını en aza indirmeye çalışan bir sezgisel yöntem geliştirilmiştir. Bu sezgisel yöntemin faydası deneysel olarak gösterilmiştir. İkincisi için, zayıf konum

k-anonimliği (anlık konum k-anonimliği sağlamak için) ve güçlü konum k-anonimliği (tarihsel konumu sağlamak için) olarak iki farklı konum k-anonimlik modeli önerilmiştir. Bu modeller, ardışık konum güncellemeleri arasındaki zamansal korelasyonları, yani hız-tabanlı saldırıları ortadan kaldırmayı amaçlamaktadır. Bu şekilde, saldırganın herhangi bir kullanıcının nerede olduğuna dair simüle edilmiş kanısının kardinalitesi asla k'dan az olmaz. İlgili algoritmalar, konum anonimliği ihlallerini tespit eder ve ilgili konum paylaşım isteklerini engeller. Sonuç olarak, kullanıcı hareket halindeyken herhangi bir zamanda güvenli bir şekilde seyahat edebilir ve konum güncellemeleri yapabilir.

Deneysel çalışmalar kapsamında, mahremiyet/fayda dengesini incelemek için kullanıcı hareketliliği farklı boyutlardaki üç gerçek açıklamalı şehir ağında simüle edilmiştir. Önerilen MCLV sezgisel yönteminin faydası deneysel olarak gösterilmiş ve efektiflik değerinin her zaman rastgele prototip seçimlerine göre istatistiksel olarak anlamlı olduğu bulunmuştur. Zayıf ve güçlü konum k-anonimliği sağlayan algoritmaların efektiflik sonuçları da sunulmuştur. Efektiflik sonuçları, mahremiyet/fayda dengesini doğrular, yani k değeri ne kadar yüksek olursa, konum güncelleme isteği engelleme oranı o kadar yüksek olur. Benzer şekilde, güçlü konum k-anonimliğin, zayıf konum k-anonimliğe kıyasla daha yüksek engelleme oranlarına neden olduğu gösterilmiştir. Ayrıca, daha hızlı hareketlerin daha yüksek blokaj oranlarına neden olacağı ikilemi de doğrulanmıştır. Çevrimiçi aşama algoritmaları, her konum güncellemesi için $O(1)$ zaman karmaşıklığına sahiptir.

Tezin ikinci bölümünde (Bölüm 4), KTS'nin tüm kullanıcılardan aldığı servis taleplerini kullanarak kullanıcılarına ilişkin ortak konumda bulunma bilgisi oluşturabileceği düşünülerek Bölüm 3'te önerilen çatıya ek olarak farklı bir saldırı modeli (ortak konumlandırma saldırısı) tanıtılmıştır. Bölüm 4'te detaylandırılan ortak konumlandırma saldırılarında, kullanıcıların konum mahremiyetini izolasyon içinde değerlendirmenin yetersiz kalacağı anlatılmış, problemle ilgili mahremiyet modeli ve saldırı modeli ayrıntılı olarak verilmiştir. Hareketlilik modeli olarak, tezin birinci bölümünde önerilen hareketlilik modeli takip edilmiştir. KTS sağlayıcısı saldırgan olduğundan dolayı, hedef bir u_1 kullanıcısının nerede olduğuna ilişkin kanısını daraltmak amacıyla ortak konumda bulunma ihtimali olan diğer kullanıcıların konum bilgilerini kullanarak u_1 kullanıcısının konumuna ilişkin çıkarım kanalları oluşturur. Çıkarım kanalları, sona erme zamanı Δt ve ortak konumda bulunma olasılığı $CLP_{u_1}(u_2)$ ile parametrize edilir ve zamansal bir boyuta da sahiptir. Geçerli çıkarım kanalları, hedef u_1 kullanıcısının konum bilgisine ilişkin bilgi taşır ve efektif belirsizlik bölgeleri yaratır. Bu efektif belirsizlik bölgeleri, KTS'nin u_1 kullanıcısının nerede olduğuna ilişkin sonraki kanılarını oluşturur. Herhangi bir sonraki kanının içerdiği belirsizlik bölgelerinin kardinalitelerinin ağırlıklı ortalaması, u_1 kullanıcısının ilgili sonraki kanı üzerinde anlık olarak anonim olabileceği düğüm sayısını gösterir. Mahremiyet modeli, konum paylaşımları sonucunda oluşan sonraki

kanıların konum k-anonimliğine dayanır. Saldırı modeli, veri alıcısının sonraki kanı modellemesine dayanır.

Çözüm yaklaşımı adanmış bir TTP'nin, KTS'nin şu anki ve sonraki kanılarını simüle etmesine dayanır. Hedef bir u_1 kullanıcısının çıkarım kanalları altında konum k-anonimliğini korumak için ortak konum k-anonimlik modeli önerilmiştir. Ortak konum k-anonimlik modeli, u_1 kullanıcısıyla ortak konumda bulunma olasılığı olan grup boyutu ile parametrize edilir. Bu model, ortak konumda bulunma olasılığı olan kullanıcılara yönelik yapılabilecek olan ortak konumlandırma saldırılarına karşı kullanıcıları korumayı amaçlamaktadır. Bu şekilde, saldırganın hedef bir u_1 kullanıcısına yönelik çıkarım kanalları altında oluşan sonraki kanılarının kardinalitesi asla k'dan az olmaz. Önerilen algoritmayı kullanan TTP, konum anonimliği ihlallerini tespit eder ve ilgili konum paylaşım isteklerini engeller. Sonuç olarak, kullanıcılar hareket halindeyken herhangi bir zamanda güvenli bir şekilde arkadaşlarıyla seyahat edebilir ve konum güncellemeleri yapabilir.

Deneysel çalışmalar kapsamında, ortak konum k-anonimliği korumaya yönelik önerilen algoritmanın mahremiyet/fayda dengesini incelemek amacıyla kullanıcı hareketliliği farklı boyutlardaki üç gerçek açıklamalı şehir ağı üzerinde simüle edilmiştir. Deneyler iki farklı senaryoda gerçekleştirilmiştir. İlk senaryoda hedef kullanıcının anonimlik seviyesi k, ACCN'i oluşturmak için kullanılan anonimlik seviyesiyle aynıdır. İkinci senaryoda ise hedef kullanıcının KTS'den daha iyi gizlenebilmek adına TTP'den daha katı bir konum mahremiyeti isteyebileceği varsayılmış ve bu nedenle TTP'den beklenilen anonimlik seviyesi, ACCN'i oluşturmak için kullanılan anonimlik seviyesinden daha düşük olarak ayarlanmıştır. Sonrasında, hedef bir u_1 kullanıcısının zayıf/güçlü konum k-anonimliğini koruyan konum güncellemeleri, TTP tarafından ortak konum k-anonimlik ihlallerine karşı kontrol edilir.

Deneysel sonuçların tümü, ortak konumlandırma saldırılarının konum anonimliğini sağlama konusunda ciddi bir problem oluşturabileceğini göstermektedir. Çıkarım kanalı parametreleri Δt ve $CLP_{u_1}(u_2)$'nin ortak konum k-anonimlik ihlali oranına etkisi deneysel olarak incelenmiş ve bu değerlerdeki artışın daha yüksek engelleme oranlarına neden olduğu gösterilmiştir. Ayrıca tüm deneyler, mahremiyet/fayda dengesini doğrular, yani hedef kullanıcının anonimlik seviyesi k ne kadar yüksek olursa, ortak konum k-anonimlik ihlali sebebiyle konum güncelleme isteği engelleme oranı o kadar yüksek olur. Önerilen algoritma her konum güncellemesinde $O(Vlog(V))$ zaman karmaşıklığına sahiptir ve toplam kullanıcı sayısından bağımsız olarak ACN'in düğüm sayısıyla yarı doğrusal olarak ölçeklenmektedir.

Sonuç olarak bu tez çalışmasında kentsel alandaki kullanıcıların saldırgan bir KTS üzerinden konum mahremiyeti korumalı konum güncellemeleri yapabilmelerini

sağlayan veri-merkezli bir çatı önerilmiştir. Farklı mahremiyet modellerine sahip olan bu çatı, kullanıcıların ilgili mahremiyet gereksinimlerine göre hizmet sağlayabilecek şekilde özelleştirilmiştir. Önerilen çatı, literatürdeki diğer yöntemlere göre ([16, 31]) kullanıcıların mahremiyet gereksinimlerinin daha basitçe, yani sadece k değerinin seçilerek belirlenmesine izin verir ve bu bağlamda kullanıcıya binen yükü (kullanıcı tarafından çok fazla parametrik değerin belirlenmesi gibi) azaltmaktadır. Önerilen algoritmalar kriptografik yöntemler kullanmamaktadır ve düşük zaman karmaşıklığına sahiptir. Bu bağlamda bu tezde önerilen konum anonimleştirme çatısının ölçeklenebilirliği yüksektir ve konum mahremiyeti sağlamak amacıyla çok fazla kullanıcısı olan KTS'ler tarafından rahat bir şekilde uygulanabilir.

Önerilen çatıda, ortak konum k-anonimlik özelliğinin anlık olarak sağlanması düşünülmüştür. İleriki bir çalışmada ortak konum k-anonimlik özelliğinin tarihsel olarak sağlanması konusuna çalışılacaktır. Ayrıca her ortak konum k-anonimlik ihlali durumu, kullanıcıların kesinlikle birlikte olduğu anlamına gelmez. Bu nedenle ve önerilen algoritmaların faydasını arttırmak için, ileriki bir çalışmada ortak konumlandırma saldırılarına hata metriği eklenmesi düşünülmektedir. Bu tezde önerilen çatıda konumlara anlamsal etiketler atfedilmediği için, küçük k değerlerinde yeterli konum tipi çeşitliliği (l-çeşitlilik) sağlanamayabilir. Fakat kullanıcılar görece büyük k değerleri seçerek konum tipi çeşitliliğini istatistiksel olarak arttırabilir.

KAYNAKLAR

[1] https://www.upguard.com/blog/biggest-data-breaches, 2021.

[2] https://fortune.com/2021/06/30/linkedin-data-theft-700-million-users-personal-information-cybersecurity, 2021.

[3] **A. Machanavajjhala and J. Gehrke and D. Kifer and M. Venkitasubramaniam**. *l*-diversity: privacy beyond *k*-anonymity. In *Proc. of the 22nd Int. Conf. on Data Engineering (ICDE'06)* (2006).

[4] **A. R. Beresford and F. Stajano**. Mix zones: user privacy in location-aware services. In *IEEE Annual Conference on Pervasive Computing and Communications Workshops, 2004. Proceedings of the Second* (March 2004), pp. 127–131.

[5] **Aggarwal, Gagan and Panigrahy, Rina and Feder, Tomás and Thomas, Dilys and Kenthapadi, Krishnaram and Khuller, Samir and Zhu, An**. Achieving anonymity via clustering. *ACM Trans. Algorithms 6*, 3 (jul 2010).

[6] **Agrawal, Rakesh and Srikant, Ramakrishnan**, t. . P. In *Proceedings of the 2000 ACM SIGMOD International Conference on Management of Data* (New York, NY, USA, 2000), SIGMOD '00, ACM, pp. 439–450.

[7] **Arain, Qasim and Memon, Imran and Liang, Deng and Memon, Muhammad and Mangi, Farman and Zubedi, Asma**. Location monitoring approach: multiple mix-zones with location privacy protection based on traffic flow over road networks. *Multimedia Tools and Applications 77* (03 2018), 5563–5607.

[8] **Ardagna, Claudio A. and Cremonini, Marco and Damiani, Ernesto and di Vimercati, Sabrina De Capitani and Samarati, Pierangela**. Location privacy protection through obfuscation-based techniques. In *21st Annual IFIP WG 11.3 Working Conference on Data and Applications Security* (July 2007).

[9] **B. Gedik and L. Liu**. Location Privacy In Mobile Systems: A Personalized Anonymization Model. In *Proc. of 25th IEEE International Conference on Distributed Computing Systems (ICDCS'05)* (2005), pp. 620–629.

[10] **Bettini, Claudio and Wang, X. Sean and Jajodia, Sushil**. Protecting privacy against location-based personal identification. In *Secure Data*

Management (Berlin, Heidelberg, 2005), W. Jonker and M. Petković, Eds., pp. 185–199.

[11] **Bonchi, F. and Saygin, Y. and Verykios, V. S. and Atzori, M. and Gkoulalas-Divanis, A. and Kaya, S. V. and Savaş, E.", editor="Giannotti, Fosca and Pedreschi, Dino**. *Privacy in Spatiotemporal Data Mining*. Berlin, Heidelberg, 2008, pp. 297–333.

[12] **C. Chow and M. F. Mokbel and W. G. Aref**. Casper*: Query Processing for Location Services without Compromising Privacy. *ACM Transactions on Database Systems (34)4* (2009).

[13] **C. Silvestri and E. Yigitoglu and M.L. Damiani and O. Abul**. SAWLnet: Sensitivity Aware Location Cloaking on Road Networks. In *Proc. of IEEE Mobile Data Management (MDM 2012)* (2012).

[14] **C-Y Chow and M. F. Mokbel**. Trajectory privacy in location-based services and data publication. *SIGKDD Explorations 13*, 1 (2011), 19–29.

[15] **D. E. O'Leary**. Knowledge discovery as a threat to database security. In *Knowledge Discovery in Databases*, G. Piatetsky-Shapiro and W. J. Frawley, Eds. AAAI/MIT Press, 1991, pp. 507–516.

[16] **E. Yigitoglu and M.L. Damiani and O. Abul and C. Silvestri**. Privacy-preserving sharing of sensitive semantic locations under road-network constraints. In *Proc. of IEEE Mobile Data Management (MDM 2012)* (2012).

[17] **Faisal Abu-Khzam and Cristina Bazgan and Katrin Casel and Henning Fernau**. Building Clusters with Lower-Bounded Sizes. In *27th International Symposium on Algorithms and Computation (ISAAC 2016)* (2016), S.-H. Hong, Ed., vol. 64, pp. 4:1–4:13.

[18] **Fan, Qi and Zhang, Dongxiang and Wu, Huayu and Tan, Kian-Lee**. A general and parallel platform for mining co-movement patterns over large-scale trajectories. *Proc. VLDB Endow. 10*, 4 (2016), 313–324.

[19] **Fuyu Liu and Kien A. Hua and Ying Cai**. Query l-diversity in location-based services. *2009 Tenth International Conference on Mobile Data Management: Systems, Services and Middleware* (2009), 436–442.

[20] **G. Ghinita and M.L. Damiani and C. Silvestri and E. Bertino**. Preventing Velocity-based Linkage Attacks in Location-Aware Applications. In *Proc. of the 17th ACM GIS* (2009).

[21] **Ghinita, Gabriel and Damiani, Maria Luisa and Silvestri, Claudio and Bertino, Elisa**. Protecting against velocity-based, proximity-based, and external event attacks in location-centric social networks. *ACM Trans. Spatial Algorithms Syst. 2*, 2 (June 2016).

[22] **Gruteser, M. and Grunwald, D.** Anonymous Usage of Location-Based Services Through Spatial and Temporal Cloaking. In *Proc. of the 1st International*

Conference on Mobile systems, Applications and Services (2003), ACM Press.

[23] **H. Kido and Y. Yanagisawa and T. Satoh**. Protection of location privacy using dummies for location-based services. In *Proc. of 21st International Conference on Data Engineering Workshops (ICDEW '05)* (2005).

[24] **J. Krumm**. A survey of computational location privacy. *Personal and Ubiquitous Computing (13)6* (2009), 391–399.

[25] **Karypis, George and Kumar, Vipin**. A fast and high quality multilevel scheme for partitioning irregular graphs. *SIAM Journal on Scientific Computing 20*, 1 (1998), 359–392.

[26] **L. Sweeney**. k-anonymity: A model for protecting privacy. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems 10*, 05 (2002), 557–570.

[27] **Lee, B. and Oh, J. and Yu, H. and Kim, J.** Protecting location privacy using location semantics. In *Proc. of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining* (2011).

[28] **M. Atallah and E. Bertino and A. Elmagarmid and M. Ibrahim and V. S. Verykios**. Disclosure limitation of sensitive rules. In *Proc. of the 1999 IEEE Knowledge and Data Engineering Exchange Workshop (KDEX'99)* (1999), pp. 45–52.

[29] **M. E. Nergiz and M. Atzori and Y. Saygin and B. Güç**. Towards trajectory anonymization: a generalization-based approach. *Transactions on Data Privacy 2*, 1 (2009), 47–75.

[30] **M. L. Damiani and C. Silvestri and E. Bertino**. Fine-Grained Cloaking of Sensitive Positions in Location-Sharing Applications. IEEE Pervasive Computing. *IEEE Pervasive Computing 10(4)* (2011), 64–72.

[31] **M. L. Damiani and E. Bertino and C. Silvestri**. The PROBE Framework for the Personalized Cloaking of Private Locations. *Transactions on Data Privacy (3)2* (2010), 123–148.

[32] **M. Terrovitis and N. Mamoulis**. Privacy preservation in the publication of trajectories. In *The Ninth International Conference on Mobile Data Management (mdm 2008)* (April 2008), pp. 65–72.

[33] **M. Xue and P. Kalnis and H.K. Pung**. Location Diversity: Enhanced Privacy Protection in Location Based Services. In *Proc. of the 4th International Symposium on Location and Context Awareness (LoCA)* (2009.).

[34] **Memon, Imran and Arain, Qasim Ali and Memon, Muhammad Hammad and Mangi, Farman Ali and Akhtar, Rizwan**. Search me if you can: Multiple mix zones with location privacy protection for mapping services. *International Journal of Communication Systems 30*, 16 (2017), e3312. e3312 IJCS-16-0125.R1.

[35] **Memon, Imran and Chen, Ling and Arain, Qasim Ali and Memon, Hina and Chen, Gencai**. Pseudonym changing strategy with multiple mix zones for trajectory privacy protection in road networks. *International Journal of Communication Systems 31*, 1 (2018), e3437. e3437 IJCS-16-0802.R2.

[36] **N. Li and T. Li and S. Venkatasubramanian**. T-Closeness: Privacy Beyond K-Anonymity And L-Diversity. In *Proc. of 23rd International Conference on Data Engineering (ICDE)* (2007), pp. 106–115.

[37] **N. Min-Allah and B. Abdullah Alahmed and E. Mohammed Albreek and L. Shabab Alghamdi and D. Abdullah Alawad and A. Salem Alharbi and N. Al-Akkas and D. Musleh and S. Alrashedb**. A survey of covid-19 contact-tracing apps. *Computers in biology and medicine* (10 2021).

[38] **Olteanu, Alexandra and Huguenin, Kévin and Shokri, Reza and Hubaux, Jean-Pierre**. Quantifying the effect of co-location information on location privacy.

[39] **OpenStreetMap contributors**. Planet dump retrieved from https://planet.osm.org . https://www.openstreetmap.org, 2017.

[40] **Osman Abul and Francesco Bonchi and Mirco Nanni**. Anonymization of moving objects databases by clustering and perturbation. *Information Systems 35*, 8 (2010), 884 – 910.

[41] **P. Samarati and L. Sweeney**. Generalizing data to provide anonymity when disclosing information. In *PODS* (1998), vol. 98, p. 188.

[42] **Palanisamy, Balaji and Liu, Ling**. Mobimix: Protecting location privacy with mix-zones over road networks. In *2011 IEEE 27th International Conference on Data Engineering* (2011), pp. 494–505.

[43] **S. O'Dea**. Smartphone users worldwide 2016-2023. https://www.statista.com/statistics/330695/number-of-smartphone-users-worldwide, 2021.

[44] **Steiniger, Stefan and Neun, Moritz and Edwardes, Alistair**. *Foundations of Location Based Services*. 01 2006.

[45] **Stenneth, Leon and Yu, Philip**. Mobile systems privacy: 'mobipriv' a robust system for snapshot or continuous querying location based mobile systems. *Transactions on Data Privacy Volume 5* (04 2012), 333–376.

[46] **Stuart A. Thompson**. The 'Holy Grail' for marketers. https://www.nytimes.com/interactive/2019/12/19/opinion/location-tracking-cell-phone.html, 2021.

[47] **Vicente, Carmen and Freni, Dario and Bettini, Claudio and Jensen, Christian**. Location-related privacy in geo-social networks. *Internet Computing, IEEE 15* (07 2011), 20 – 27.

[48] **Virrantaus, K. and Markkula, J. and Garmash, A. and Terziyan, V. and Veijalainen, J. and Katanosov, A. and Tirri, H.** Developing gis-supported location-based services. In *Proceedings of the Second International Conference on Web Information Systems Engineering* (2001), vol. 2, pp. 66–75 vol.2.

[49] **Weiler, Michael and Schmid, Klaus Arthur and Mamoulis, Nikos and Renz, Matthias**. Geo-social co-location mining. In *Second International ACM Workshop on Managing and Mining Enriched Geo-Spatial Data* (New York, NY, USA, 2015), GeoRich'15, Association for Computing Machinery, p. 19–24.

[50] **Wu, Songyang and Xu, Wenju and Hong, Zhiyong and Duan, Pu and Zhang, Benyu and Hu, Yupu and Wang, Baocang**. Updatable privacy-preserving k-nearest neighbor query in location-based service. *Peer-to-Peer Networking and Applications* (01 2022).

[51] **Xinxin Liu and Han Zhao and Miao Pan and Hao Yue and Xiaolin Li and Fang, Yuguang**. Traffic-aware multiple mix zone placement for protecting location privacy. In *2012 Proceedings IEEE INFOCOM* (2012), pp. 972–980.

[52] **Xu, Zhikai and Zhang, Hongli and Yu, Xiangzhan**. Multiple mix-zones deployment for continuous location privacy protection. In *2016 IEEE Trustcom/BigDataSE/ISPA* (2016), pp. 760–766.

[53] **Yang, Manxiang and Ye, Baopeng and Chen, Yuling and Li, Tao and Yang, Yixian and Qian, Xiaobin and Yu, Xiaomei**. A trusted de-swinging k-anonymity scheme for location privacy protection. *Journal of Cloud Computing 11* (01 2022).

[54] **Yarovoy, Roman and Bonchi, Francesco and Lakshmanan, Laks V. S. and Wang, Wendy Hui**. Anonymizing moving objects: How to hide a mob in a crowd? In *Proceedings of the 12th International Conference on Extending Database Technology: Advances in Database Technology* (New York, NY, USA, 2009), EDBT '09, ACM, pp. 72–83.

[55] **Zhang, Haitao AND Wu, Chenxue AND Chen, Zewei AND Liu, Zhao AND Zhu, Yunhong**. A novel on-line spatial-temporal k-anonymity method for location privacy protection from sequence rules-based inference attacks. *PLOS ONE 12*, 8 (08 2017), 1–32.