



## SOĞUK ZİNCİR LOJİSTİĞİNDE GEZGİN SATIN ALICI PROBLEMİ İÇİN İKİ AŞAMALI BİR ÇÖZÜM YÖNTEMİ

İlker KÜÇÜKOĞLU\*

Bursa Uludağ Üniversitesi, Mühendislik Fakültesi, Endüstri Mühendisliği Bölümü, Bursa, Türkiye

### Anahtar Kelimeler

*Gezgin Satın Alıcı Problemi,  
Soğuk Zincir Lojistiği,  
Matematiksel Modelleme,  
Sezgisel Yöntemler.*

### Öz

Bu çalışma, soğuk zincir lojistiğinde bozulabilir gıdaların satın alınması işlemini dikkate alarak literatürde iyi bilinen gezgin satın alıcı probleminin genişletilmiş bir halini sunmaktadır. Taşıma işlemlerinin bozulabilir ürün şartlarına dayalı gerçekleştirildiği bu problem, soğuk zincir lojistiğinde gezgin satın alıcı problemi (SZL-GSAP) olarak adlandırılmıştır. SZL-GSAP, sıcaklık kontrollü bir araç ile taşınması gereken çeşitli ürün taleplerinin belirli sayıda marketlerden satın alınması işlemini dikkate almaktadır. Problemin amacı, satın alıcı için sıcaklık kontrollü aracın enerji maliyetinin, ürün bozulma maliyetinin ve satın alma maliyetinin toplamını minimize edecek satın alma ve rota planını elde etmektir. Problem, doğrusal olmayan bir karma tamsayı programlama modeli olarak formüle edilmiş ve iki aşamalı bir yöntem kullanılarak çözülmüştür. Çözüm yönteminin ilk aşamasında, bir çözüm kurucu yöntem kullanılarak problem için kaba bir sonuç elde edilmektedir. Daha sonra ilk aşamada elde edilen çözüm, GUROBI çözücüsü kullanılarak belirli bir zaman kısıtı içerisinde geliştirilmektedir. Sayısal uygulamalarda, önerilen çözüm yönteminin performansı, farklı büyüklükte örnekleri içeren bir problem seti üzerinde analiz edilmiştir.

## A TWO-PHASE SOLUTION APPROACH FOR THE TRAVELING PURCHASER PROBLEM IN COLD CHAIN LOGISTICS

### Keywords

*Traveling Purchaser Problem,  
Cold Chain Logistics,  
Mathematical Modelling,  
Heuristic Methods.*

### Abstract

This paper introduces an extension of the well-known traveling purchaser problem in the literature by taking into account the procurement of perishable foods in cold chain logistics. The problem in which transportation operations are carried out based on the perishable product conditions is called the traveling purchaser problem in cold chain logistics (TPP-CCL). The TPP-CCL considers a set of perishable food demands to be purchased from a number of markets, where the foods have to be transported by a temperature-controlled vehicle. The aim of the problem is to find the best procurement and route plan for the purchaser that minimizes the total energy cost of the vehicle, damage cost of foods and purchasing cost. The problem is formulated as a nonlinear mixed-integer programming model and solved by using a two-phase solution procedure. In the first phase of the solution methodology, a rough solution is obtained through a solution construction method. Then the solution observed in the first phase is improved by using the GUROBI solver, where the solver is terminated at the end of a specific time limit. In the computational studies, the performance of the proposed solution methodology is analyzed on a benchmark problem set including different sized instances.

### Alıntı / Cite

Küçüköğlü, İ., (2022). Soğuk Zincir Lojistiğinde Gezgin Satın Alıcı Problemi İçin İki Aşamalı Bir Çözüm Yöntemi, Mühendislik Bilimleri ve Tasarım Dergisi, 10(3), 929-942.

\* İlgili yazar / Corresponding author: ikucukoglu@uludag.edu.tr, +90-224-294-2091

Yazar Kimliği / Author ID (ORCID Number)	Makale Süreci / Article Process	
İ. Küçüköğlü, 0000-0002-5075-0876	<b>Başvuru Tarihi / Submission Date</b>	20.12.2021
	<b>Revizyon Tarihi / Revision Date</b>	14.05.2022
	<b>Kabul Tarihi / Accepted Date</b>	14.05.2022
	<b>Yayın Tarihi / Published Date</b>	30.09.2022

## 1. Giriş (Introduction)

Gezgin satın alıcı problemi (GSAP), literatürde iyi bilinen gezgin satıcı probleminin genelleştirilmiş bir hali olup birçok çalışmada farklı varsayımlar altında dikkate alınmıştır (Manerba vd., 2017). Temel GSAP'de bir gezgin satın alıcı, belirli ürün taleplerini karşılamak üzere farklı lokasyonlarda bulunan marketleri ziyaret etmektedir. Her markette bulunan ürün çeşidi ve miktarı birbirinden farklı olmakta ve birim ürün satış fiyatı markete göre değişiklik göstermektedir. Gezgin satın alıcının amacı, ürün taleplerini minimum maliyet ile karşılayacak rota ve ürün satın alma planını oluşturmaktır. Gezgin satıcı probleminden farklı olarak GSAP'de ziyaret edilecek olan marketlerin seçimi ve bu marketlerden alınacak olan ürün miktarlarının belirlenmesi gerekmektedir. Bu temel karar alma süreçlerine ek olarak, GSAP farklı kısıtlar ile genişletilerek birçok gerçek hayat uygulamasına adapte edilmiştir.

GSAP'nin önemli bir karar alma süreci olan taşıma göz önüne alındığında, günümüz lojistik faaliyetlerinde taşıma işlemlerinin önemli bir kısmının soğuk zincir lojistiği kapsamında yapıldığı bilinmektedir (Awad vd., 2021; Oswald ve Stirn, 2008). Soğuk zincir lojistiğinde, sıcaklık değişimine bağlı olarak bozulabilir ürünlerin sıcaklık kontrollü araçlar ile yapılması gerekmektedir. Standart taşıma işlemlerinden farklı olarak soğuk zincir lojistiğinde aracın ısı kontrolü için harcadığı yakıtta ait maliyetler ve ürünlerin bozulma maliyetleri de dikkate alınması gereken önemli bir faktördür (Zhang vd., 2019; Liu vd., 2020). Fakat bu alanda yapılmış mevcut çalışmalar genel olarak bozulabilir ürünlerin sıcaklık kontrollü araçlar ile belirli lokasyonlara dağıtımını dikkate almaktadır. Taşıma faaliyetleri açısından dağıtım işlemlerine ek olarak soğuk zincir lojistiğine ait gerçek hayat uygulamalarında bozulabilir ürünlerin belirli üretici noktalarından tedarik edilmesine bağlı taşıma da gerçekleşmektedir. Özellikle günlük olarak üretilen süt ve süt ürünleri, hayvansal gıdalar, tarım ürünleri vb. ürünlerin tedarik edilmesi aşamasında ürün kalitesi dikkate alınarak taşımaların belirli bir zaman içerisinde yapılması gerekmektedir. Bu tür ürünlerin satın alımı aşamasında ürün maliyetleri ve taşıma maliyetlerine ek olarak taşımalardan kaynaklı ürün bozulma maliyetlerinin de dikkate alınması gerekmektedir. Bozulabilir ürünlerin satın alınmasına ilişkin bu kısıtların GSAP kapsamında dikkate alınması, toplam taşıma ve ürün maliyetlerinin en aza indirilebilmesi açısından önemli bir araştırma konusu niteliği taşımaktadır.

Yapılan çalışmada, GSAP soğuk zincir lojistiğine ait uygulamalara yönelik genişletilmiş olup bozulabilir ürünlerin taşımalarına ilişkin kısıtlar probleme entegre edilmiştir. Soğuk zincir lojistiğinde gezgin satın alıcı problemi (SZL-GSAP) olarak adlandırılmış olan problemde bozulabilir ürünlerin farklı lokasyonlarda bulunan marketlerden satın alınmasına ilişkin karar alma süreci dikkate alınmaktadır. Ürün taşımalarının sıcaklık kontrollü bir araç ile yapıldığı varsayılan problemde ürün satın alma ve araç dolaşım maliyetlerine ek olarak ürünlerin zamana bağlı bozulmalarından kaynaklı maliyetler de göz önünde bulundurulmaktadır. Bu kapsamda problemin amacı ürün satın alma, araç enerji maliyeti (dolaşım ve bekleme) ve zamana bağlı ürün bozulma maliyetlerinin toplamını minimize etmektir. Bu varsayımlar altında SZL-GSAP doğrusal olmayan karışık tamsayı matematiksel programlama modeli olarak formüle edilmiş olup problemin çözümü için iki aşamalı bir çözüm yöntemi önerilmiştir. Çözüm yönteminin ilk aşamasında probleme özgü geliştirilmiş olan bir çözüm kurucu sezgisel yöntem ile gezgin satın alıcı için kaba bir satın alma ve rota planı oluşturulmaktadır. İkinci aşamada ise ilk aşamada oluşturulmuş olan çözüm, kesin çözüm veren dal-kesim (branch-and-cut) yöntemini kullanan GUROBI çözücüsü ile belirli bir çözüm zamanı limiti ile geliştirilmektedir.

Yapılan çalışma kapsamında, GSAP'ye ilişkin literatürde yer alan mevcut çalışmalar ikinci bölümde verilmiştir. Soğuk zincir lojistiğine dayalı gezgin satın alıcı problemine ilişkin problem tanımı ve matematiksel formülasyonu üçüncü bölümde verilmiştir. Dördüncü bölümde, problemin çözümüne ilişkin geliştirilmiş olan çözüm yaklaşımı tanımlanmıştır. Geliştirilmiş olan çözüm yaklaşımının geçerliliğinin ve performansının analiz edilmesine ilişkin sayısal çalışmalar beşinci bölümde verilmiştir. Altıncı bölümde ise çalışmaya ilişkin sonuç ve olası gelecek çalışma konuları yer almaktadır.

## 2. Kaynak Araştırması (Literature Survey)

GSAP, dikkate alınmış olduğu karar alma süreçleri açısından satın alma ve rotalama planlarını bir arada içermesi nedeniyle son yıllarda birçok akademik çalışmada ve gerçek hayat uygulamasında çalışılan bir problem olmuştur. Bu problem temel alınarak depolama, ulaştırma, telekomünikasyon şebeke tasarımı, çizelgeleme vb. birçok problem matematiksel olarak formüle edilmiştir (Manerba vd., 2017). Bu kapsamda GSAP için farklı varyasyonlar

öne sürülmüştür. Bu varyasyonlardan literatürde en sık çalışılmış olanı kapasitesi GSAP'dir (Boctor vd., 2003). Kapasitesiz GSAP'de her ürün çeşidinin marketlerde talebi karşılayabilecek yeterli miktarın var olduğu varsayılmaktadır. Bu varsayım altında ürün talebinin tek bir marketten karşılanması söz konusu olduğundan problem, ürün talep miktarlarını bir olarak kabul ederek daha basit hale indirgenebilmektedir. Literatürde GSAP için sıklıkla dikkate alınan diğer bir varyasyon ise iki amaçlı GSAP'dir. İki amaçlı GSAP'de model iki farklı amaç fonksiyonunu (örn: satın alma maliyeti ve dolaşım maliyeti) dikkate almaktadır. Matematiksel model, amaç fonksiyonlarını ayırık olarak ele alarak veya her bir amaç fonksiyonuna ağırlık vererek çözülmektedir (Riera-Ledesma ve Salazar-González, 2005; Almeida vd., 2012). GSAP'de ürün satın alma işlemlerine ilişkin önerilmiş olan diğer bir varyasyon ise dinamik GSAP'dir. Dinamik GSAP'de marketlerde yer alan ürün miktarlarının zamana bağlı olduğu ve zaman ilerledikçe marketlerde yer alan ürün miktarlarının azaldığı varsayılmaktadır (Angelelli vd., 2017). Dinamik GSAP'ye benzer yapıda çalışılan diğer bir GSAP varyasyonu ise ürün fiyatlarının olasılık dağılımlarına göre belirlendiği stokastik GSAP'dir. Stokastik GSAP'de gezgin satın alıcı belirli bir ürüne ait fiyatı marketi ziyaret ettiği zaman öğrenebilmektedir ve satın alma işlemini önerilmiş olan fiyata göre yapmaktadır (Kang ve Ouyang, 2011). Satın alma işlemlerine ait çalışılmış başka bir GSAP varyasyonu ise bütçe kısıtlı GSAP'dir. Problemden toplam satın alma maliyetleri için bir bütçe limiti belirlenmektedir ve satın alınan ürünlerin toplam maliyeti bu limiti aşmamaktadır (Mansini ve Tocchella, 2009). Literatürde sıklıkla çalışılan diğer bir varyasyon ise çok amaçlı GSAP olmuştur. Çok amaçlı GSAP'de satın alma işlemleri için belirli bir menzil veya kapasiteye sahip birden fazla araç kullanılmaktadır (Choi ve Lee, 2011; Riera-Ledesma ve Salazar-González, 2012). Yakın zamanda yapılan çalışmalarda ise çevresel kaygılar dikkate alınarak GSAP, gezgin satın alıcının dolaşımında sera gazı salınımlarını en aza indirecek şekilde genişletilmiştir (Hamdan vd., 2017; Cheaitou vd., 2020).

GSAP, gezgin satıcı problemi, tesis yer seçimi problemi ve küme kaplama problemine ait kararların tümünü içermesi nedeniyle çözüm zorluğu açısından NP-Zor sınıfta yer aldığı bilinmektedir (Manerba vd., 2017; Mansini ve Tocchella, 2009). Bu nedenle literatürde GSAP'nin çözümü için bir çok çözüm yaklaşımı geliştirilmiştir. Bu çözüm yaklaşımları temel olarak optimum sonucu garanti eden kesin çözüm yöntemleri ve optimum sonuca yakın sonucu bulmayı hedefleyen sezgisel yöntemler olarak iki gruba ayrılmaktadır. Kesin çözüm veren yöntemler arasında dal-sınır, Laporte ve dal-kesim ve dal-ücret algoritmaları sıklıkla kullanılan yöntemler arasındadır (Singh ve van Oudheusden, 1997; Laporte vd., 2003; Riera-Ledesma ve Salazar-González, 2006; Bianchessi vd., 2014; Gendreau vd., 2016; Bianchessi vd., 2021). Bu yaklaşımlara ek olarak Gouveia vd. (2011) ve Kang ve Ouyang (2011) tarafından dinamik programlama ve Cambazard ve Penz (2012) tarafından kısıt programlama yaklaşımları kullanılmıştır. Optimum sonucu bulmayı garanti eden bu yöntemler birçok GSAP varyasyonu için çözüm yöntemi olarak dikkate alınmıştır. Diğer yandan daha kısa işlem zamanları ile optimum veya optimum sonuca yakın çözümler üretebilmek amacıyla GSAP'nin çözümü için birçok sezgisel yöntem geliştirilmiştir. Bu kapsamda yapılan ilk çalışmalarda genelleştirilmiş tasarruf algoritması (Golden vd., 1981), tur azaltma algoritması (Ong, 1982) ve ürün ekleme algoritması (Pearn, 1991) olarak adlandırılmış çözüm kurucu yöntemler geliştirilmiştir. Bu yöntemler GSAP için etkin çözüm üretebilen çözüm kurucu yöntemler olup birçok GSAP varyantının çözümü için geliştirilerek kullanılmıştır (Boctor vd., 2003; Pearn ve Chien, 1998; Teeninga ve Volgenant, 2004). Çözüm kurucu yöntemlere ek olarak daha etkin sonuçlar üretebilmek için meta-sezgisel yöntemler de GSAP'nin çözümü için kullanılmıştır. Bu çalışmalara Voß (1996) tarafından geliştirilmiş olan tabu arama algoritması ve tavlama benzetimi algoritması, Bontoux ve Feillet (2008) tarafından geliştirilmiş olan karınca kolonisi algoritması, Palomo-Martínez ve Salazar-Aguilar (2019) tarafından geliştirilmiş olan değişken komşu çözüm arama algoritması ve Cheaitou vd. (2020) tarafından geliştirilmiş olan genetik algoritma örnek olarak gösterilebilir.

GSAP üzerine yapılmış çalışmalara ek olarak soğuk zincir lojistiğinde rota optimizasyonu konusunda yapılmış olan mevcut çalışmalar dikkate alındığında birçok çalışma, belirli lokasyonlarda bulunan müşterilere bozulabilir ürünlerin merkezi bir depodan dağıtımına ilişkin problemi dikkate almaktadır. Soğuk Zincir Lojistiğinde Araç Rotalama Problemi (SZL-ARP) olarak adlandırılan bu problem, yapılan çalışmalarda farklı kısıtlar dahilinde genişletilmiştir. Hsu vd. (2007) yaptıkları çalışmada SZL-ARP'yi zaman penceresi kısıtları ile genişletmiş ve problemin çözümü için zaman odaklı bir en yakın komşu algoritması geliştirmişlerdir. Chen vd. (2009) tarafından yapılan çalışmada ise bozulabilir ürünlerin dağıtımında araç rotalama problemi ve üretim çizelgeleme problemi bir arada dikkate alınmıştır. Amorim vd. (2014) SZL-ARP'yi farklı soğutma özelliklerine ve taşıma kapasitelerine sahip heterojen yapıda bir filo kapsamında genişletmiştir. Yazarlar problemin çözümü için uyarlanabilir geniş komşu arama algoritması geliştirmişlerdir. Wang ve Wen (2020) SZL-ARP'yi benzer varsayımlar altında dikkate almış ve problemin çözümü için uyarlanabilir bir genetik algoritma geliştirmişlerdir. Chen vd. (2019) tarafından yapılan çalışmada ise farklı ürün gruplarına yönelik saklama koşulu sağlayan araçlar dikkate alınmıştır. Li vd. (2019) ve Liu vd. (2020) bozulabilir ürün dağıtımını operasyonlarını çevresel faktörleri de dikkate alarak yeşil araç rotalama problemi kapsamında planlamışlardır. Çevresel faktörlerin de dikkate alındığı SZL-ARP'nin çözümü için Li vd. (2019) parçacık sürü optimizasyonu algoritmasını, Liu vd. (2020) ise tavlama benzetimi algoritmasını kullanmıştır. Song vd. (2020) SZL-ARP'yi araçların toplam enerji tüketim değerlerini minimize edecek şekilde formüle etmiş ve problemin çözümü için bir yapay balık sürüsü algoritması geliştirmişlerdir. Qi ve Hu (2020)

tarafından yapılan çalışmada ise taşıma ve ürün bozulma maliyetleri trafik yol koşulları dikkate alınarak hesaplanmıştır. Problemin çözümü için yazarlar karınca kolonisi algoritmasını kullanmışlardır.

GSAP ve SZL-ARP üzerine bugüne kadar yapılmış olan mevcut çalışmalar dikkate alındığında soğuk zincir lojistiğine ait rota optimizasyonu genel olarak bozulabilir ürünlerin dağıtım işlemine yönelik çalışılmış olup ürünlerin satın alınmasını inceleyen bir çalışmaya rastlanmamıştır. Bu kapsamda yapılan çalışmanın literatüre iki önemli katkısı bulunmaktadır: (1) GSAP soğuk zincir lojistiğine ait bozulabilir ürünlerin taşımalarına dayalı kısıtlar altında ilk olarak incelenmiş ve bu kapsamda GSAP için yeni bir varyasyon önerilmiştir. Diğer yandan SZL-ARP'de genel olarak bozulabilir ürünlerin dağıtım işlemleri dikkate alınırken SZL-GSAP'de bozulabilir ürünlerin satın alınarak tedarik edilmesine ilişkin karar alma süreci dikkate alınmaktadır. (2) Önerilmiş olan çözüm yaklaşımı sezgisel ve kesin çözüm veren iki önemli yöntemi iki aşamalı dikkate alarak gerçek hayat uygulamalarına yönelik bir yaklaşım sunmaktadır. Geliştirilmiş olan bu çözüm yaklaşımı farklı boyutlarda birçok örnek problem üzerinde test edilerek performansı analiz edilmiştir.

### 3. Problem Tanımı ve Matematiksel Modeli (Problem Definition and Model Formulation)

SZL-GSAP, klasik gezgin satın alıcı probleminin genişletilmiş bir varyasyonu olup satın alma işlemlerinde soğuk zincir ürün taşımalarında ortaya çıkabilecek ürün bozulmalarına ait maliyetler ek olarak dikkate alınmaktadır. SZL-GSAP'de farklı tipte ürünlere ait taleplerin farklı lokasyonlarda bulunan marketlerden satın alınması gerekmektedir. Satın alınacak ürünler iklim koşullarına göre bozulabilir yapıda olup marketlerden alınan ürünler soğutucu özelliğe sahip bir araçta taşınmaktadır. Aracın lokasyonlar arası ulaşım maliyetine ek olarak market noktalarında beklediği sürece ürünleri soğutmak için belirli bir yakıt harcadığı dikkate alınmaktadır. Gezgin satın alıcı, turuna depo noktasından başlamakta ve satın alma işlemlerini bitirdikten sonra depo noktasına geri dönmektedir. Gezgin satın alıcının rotasında her bir market en fazla bir defa ziyaret edilebilmektedir. Her markette bulunan ürün çeşidi ve miktarı farklı olabilmektedir. Ürünlere ait talep miktarları karşılanmak zorunda olup bir ürün için marketlerde bulunan toplam ürün miktarının ilgili ürün talep miktarından fazla olduğu varsayılmaktadır. Problemden amaç, ürünlerin toplam satın alma maliyetlerine ek olarak aracın taşıma ve bekleme anında harcadığı yakıt maliyetlerini minimize etmektir.

Yukarıda tanımlanmış olan SZL-GSAP'ye ait matematiksel model, literatürde yer alan temel GSAP formülasyonu dikkate alınarak ürün taşıma sürelerini ve buna bağlı olarak ürün bozulma maliyetlerini hesaplayacak şekilde geliştirilmiştir. Ürünlerin bozulma maliyetleri, Liu vd. (2020) ve Qi ve Hu (2020) tarafından geliştirilmiş kısıtlara benzer şekilde, ürünlerin taşıma süreleri ve ziyaret edilen noktalarda bekleme süreleri üzerinden hesaplanmıştır. Modelde sıcaklık kontrollü araçların seyir halinde harcadıkları yakıt maliyetine ek olarak market noktalarında beklemeden kaynaklı soğutma maliyetleri de ek olarak dikkate alınmıştır. Bu varsayımlara göre SZL-GSAP için geliştirilmiş olan doğrusal olmayan karışık tamsayılı programlama modeli aşağıda verilmiştir.

Parametreler;

$K$	Ürünler kümesi
$M$	Market noktaları kümesi
$M_k$	$k$ ürününün mevcut olduğu marketler kümesi ( $M_k \subseteq M$ ), $k \in K$
$\{0\}$	Depo noktası
$V$	Depo ve marketler noktaları kümesi ( $M \cup \{0\}$ )
$d_k$	$k$ ürününe ait talep miktarı, $k \in K$
$q_{ik}$	$i$ marketinde mevcut olan $k$ ürünü miktarı, $k \in K, i \in M_k$
$p_{ik}$	$i$ marketinde $k$ ürününün fiyatı, $k \in K, i \in M_k$
$c_{ij}$	$i$ noktasından $j$ noktasında olan taşıma maliyeti, $i, j \in V$
$t_{ij}$	$i$ noktasından $j$ noktasında olan ulaşım süresi, $i, j \in V$
$h_i$	$i$ marketinde birim ürün başına satın alma süresi; $i \in M$
$\theta_1$	Ürünlerin taşıma esnasında bir birim zamanda bozulma katsayısı
$\theta_2$	Ürünlerin market noktalarında araç içerisinde bekleme esnasında bir birim zamanda bozulma katsayısı
$\vartheta$	Market noktalarında aracın bir birim zamanda ürünleri soğutma maliyeti
$\gamma$	Büyük bir sayı

Karar Değişkenleri;

$x_{ij}$	Eğer satın alıcı $i$ noktasından $j$ noktasına giderse 1, aksi halde 0; $i, j \in V, i \neq j$
$y_i$	Eğer market $i$ , satın alıcı tarafından ziyaret edilirse 1, aksi halde 0; $i \in M$
$z_{ik}$	$i$ marketinden satın alınan $k$ ürünü miktarı; $k \in K, i \in M_k$
$r_i$	Satın alıcının $i$ noktasına varış zamanı; $i \in V$
$s_i$	Satın alıcının $i$ marketinde harcadığı zaman; $i \in M$
$l_i$	Aracın $i$ marketine varmadan önce içerisinde yer alan ürünlerin toplam satın alma maliyeti; $i \in M$

Model;

$$\text{Min} \sum_{i \in V} \sum_{j \in V} c_{ij} x_{ij} + \sum_{k \in K} \sum_{i \in M_k} p_{ik} z_{ik} + \sum_{k \in K} \sum_{i \in M_k} \theta_1 p_{ik} z_{ik} (r_0 - r_i) + \sum_{i \in M} \theta_2 l_i s_i + \sum_{i \in M} \vartheta s_i \quad (1)$$

S.t.

$$\sum_{i \in M_k} z_{ik} = d_k \quad k \in K \quad (2)$$

$$z_{ik} \leq q_{ik} y_i \quad k \in K, \quad i \in M_k \quad (3)$$

$$\sum_{\substack{j \in V \\ i \neq j}} x_{ij} = y_i \quad i \in M \quad (4)$$

$$\sum_{\substack{i \in V \\ i \neq j}} x_{ij} = y_j \quad j \in M \quad (5)$$

$$\sum_{i \in M} x_{i0} = 1 \quad (6)$$

$$\sum_{j \in M} x_{0j} = 1 \quad (7)$$

$$t_{0i} \leq r_i + \gamma(1 - x_{0i}) \quad i \in M \quad (8)$$

$$r_i + s_i + t_{ij} \leq r_j + \gamma(1 - x_{ij}) \quad i \in M, \quad j \in V, \quad i \neq j \quad (9)$$

$$s_i = \sum_{k \in K} h_i z_{ik} \quad i \in M \quad (10)$$

$$l_i + \sum_{k \in K} p_{ik} z_{ik} \leq l_j + \gamma(1 - x_{ij}) \quad i, j \in M, \quad i \neq j \quad (11)$$

$$x_{ij} \in \{0,1\} \quad i, j \in V, \quad i \neq j \quad (12)$$

$$y_i \in \{0,1\} \quad i \in M \quad (13)$$

$$z_{ik} \geq 0 \quad k \in K, \quad i \in M_k \quad (14)$$

$$r_i \geq 0 \quad i \in V \quad (15)$$

$$s_i, l_i \geq 0 \quad i \in M \quad (16)$$

Geliştirilmiş olan matematiksel modelde denklem (1); toplam taşıma maliyetini, ürünlerin satın alma maliyetini, ürünlerin taşıma ve beklemlerden kaynaklı bozulma maliyetlerini ve aracın soğutma maliyetlerini minimize edecek amaç fonksiyonunu ifade etmektedir. Kısıt (2) her ürün tipine ait talep miktarının karşılanmasını sağlamaktadır. Kısıt (3), bir marketten alınabilecek  $k$  ürününün miktarını ilgili markette mevcut olan miktar ile sınırlamaktadır. Kısıt (3) aynı zamanda bir marketten ürün satın alınması durumunda ilgili marketin satın alıcı tarafından ziyaret edilmesini sağlamaktadır. Kısıt (4) ve (5), ürün satın alınan her markete sadece bir defa ziyaret edilmesini garanti etmektedir. Kısıt (6) ve (7), gezgin satın alıcının turuna depo noktasından başlamasını ve satın alma işlemi bittiğinde depoya geri dönmesini sağlamaktadır. Kısıt (8) ve (9), satın alıcının turunda ziyaret ettiği noktalara varış zamanını hesaplamaktadır. Kısıt (10) ziyaret edilen marketlerde satın alıcının geçirdiği süreyi hesaplamaktadır. Kısıt (11) satın alıcının turunda ziyaret edilen marketlere varış anında araç içerisinde yer alan ürünlerin toplam satın alma maliyetini hesaplamaktadır. Kısıt (12)-(16) modele ait karar değişkenlerini tanımlamaktadır.

#### 4. Çözüm Yöntemi (Solution Methodology)

GSAP, gezgin satıcı probleminin genelleştirilmiş bir varyasyonu olup problemin  $NP$ -Zor yapıya sahip olması nedeniyle çözümü büyük boyutlu problemlerde oldukça zor olduğu bilinmektedir. SZL-GSAP, GSAP'de yer alan temel varsayımlara ek olarak soğuk zincir lojistiğinde bozulabilen ürünlerin taşınmasına ilişkin varsayımları da dikkate almaktadır ve bu varsayımlar ile problem doğrusal olmayan ifadeleri içeren bir amaç fonksiyonu ile formüle edilebilmiştir. Bu nedenle SZL-GSAP'nin çözüm zorluğu GSAP'ye daha fazla olacağı ifade edilebilir.

Günümüzde birçok ticari çözücünün konveks (convex) veya konveks olmayan (non-convex) özelliğe sahip doğrusal olmayan matematiksel modeller için optimum sonucu garanti eden veya global optimuma yakın çözümler üretebildiği bilinmektedir. Fakat SZL-GSAP için yapılan ön çalışmalarda bu tür çözücülerin kısa süreler içerisinde uygun bir çözüme dahi erişemediği tespit edilmiştir. Bu nedenle bu çalışmada, SZL-GSAP'nin çözümü için iki aşamalı bir yaklaşım önerilmiştir. Çözüm yaklaşımının ilk aşamasında başlangıç çözüm algoritması (BÇA) olarak adlandırılmış olan bir çözüm kurucu yöntem kullanılarak SZL-GSAP için kaba bir çözüm üretilmektedir. Elde edilen bu çözüm ikinci aşamada GUROBI çözücüsü kullanılarak belirli bir zaman kısıtı dahilinde iyileştirilmektedir.

BÇA, Boctor vd. (2003) tarafından kapasiteli GSAP'nin çözümüne yönelik geliştirilmiş olan ürün ekleme sezgisel (commodity adding heuristic - CAH) yönteminin SZL-GSAP için genişletilmiş halidir. BÇA satın alınacak ürünleri rassal bir sırada seçerek her adımda seçilen ürünün talebini en düşük birim maliyet artışına neden olacak marketi seçerek karşılamaktadır. Algoritmada kullanılan notasyonlar;

$X$	Gezgin satın alıcının rotası
$M^V$	Ziyaret edilen marketler kümesi
$M^{UV}$	Ziyaret edilmemiş marketler kümesi
$K^P$	Talebi karşılanmış ürünler kümesi
$K^{NP}$	Talebi karşılanmamış ürünler kümesi
$d_k^U$	$k$ ürününün karşılanmayan talep miktarı, $k \in K$

BÇA'nın başlangıcında ilk olarak  $X$ ,  $M^V$  ve  $K^P$  boş kümeler olup  $M^{UV}$  bütün marketleri ve  $K^{NP}$  bütün ürünleri içermektedir. Ek olarak her bir  $k$  ürünü için  $d_k^U = d_k$  olarak belirlenmektedir. Arından algoritmanın her adımında  $K^{NP}$  listesi içerisinde bir  $k$  ürünü rassal olarak seçilmektedir. Seçilen  $k$  ürünü için  $M^V$  ve  $M^{UV}$  içerisinde  $k$  ürünü mevcut olan tüm  $i$  marketleri için birim ürün başına maliyet artış değeri  $uc_{ik}$  hesaplanmaktadır. Mevcut çözüme ait toplam maliyetin  $f$  ve  $k$  ürününün alternatif bir  $i$  marketinden satın alınması durumunda elde edilecek yeni çözümün toplam maliyetinin  $f'$  olması durumunda  $uc_{ik} = (f' - f) / \min\{q_{ik}, d_k^U\}$  olarak hesaplanmaktadır.  $f$  değerinin hesaplanmasında denklem (1)'de belirtilmiş olan aracın taşıma maliyeti, ürünlerin satın alma maliyeti, ürünlerin taşıma ve beklemelerinden kaynaklı bozulma maliyeti ve aracın soğutma maliyeti dikkate alınmaktadır.  $i$  marketinin  $M^{UV}$  kümesi içerisinde yer alması durumunda ilgili market,  $X$  içerisinde en düşük maliyet artışına neden olacak pozisyona yerleştirilmektedir.  $X = \{\emptyset\}$  ise gezgin satın alıcının rotası direkt olarak  $X = \{0, i, 0\}$  olarak belirlenir. Aksi halde  $i$  marketinin her bir  $\{j, j + 1\}$  lokasyonu arasına yerleştirilmesi durumunda elde edilen toplam maliyet üzerinden en düşük maliyetli çözüm belirlenmektedir.  $i$  marketinin mevcut rotada bulunan marketler kümesi  $M^V$  içerisinde bulunması durumunda ise  $X$  rotasında bir değişiklik olmayacaktır. Dikkate alınan  $k$  ürünü için tüm alternatif marketlerin değerlendirilmesi durumunda en düşük birim maliyet artışına sahip  $i^*$  marketi  $i^* = \operatorname{argmin}_{i \in M^V \cup M^{UV}} \{uc_{ik}\}$  olarak belirlenmektedir. Seçilen  $i^*$  marketinden  $\min\{q_{i^*k}, d_k^U\}$  miktarında  $k$  ürünü satın alınarak  $K^P$ ,  $M^V$ ,  $M^{UV}$ ,  $K^P$ ,  $K^{NP}$  ve  $d_k^U$  güncellenmektedir. Seçilen  $k$  ürünü için  $d_k^U = 0$  oluncaya kadar marketler yukarıda belirtildiği gibi değerlendirilerek satın alma işlemine devam edilmektedir. Algoritma bütün ürünlerin talebinin karşılanması durumunda sonlanmaktadır. BÇA'ya ait sözde kod Şekil 1'de verilmiştir.

Çözüm yaklaşımının ikinci aşamasında, BÇA ile elde edilen çözüm (çözüme ait rota bilgisi  $X$  ve ürün satın alma bilgisi  $z_{ik}$ ) GUROBI çözücüsüne bir başlangıç çözümü olarak verilmektedir. Bu çözüm GUROBI çözücüsü ile kullanıcı tarafından belirlenen bir zaman kısıtı boyunca iyileştirilmektedir. Belirlenmiş olan çözüm zamanı sonunda probleme ait optimum çözüm veya bir üst sınır değer elde edilmektedir. Elde edilen sonuç, çözüme ait optimalite aralığı (%Gap) üzerinden değerlendirilebilmektedir.

```

1:  $X = \emptyset, M^V = \emptyset, M^{UV} = M, K^P = \emptyset, K^{NP} = K$ 
2:  $d_k^U = d_k, k \in K$ 
3:  $f = 0$ 
4: Do
5:    $K^{NP}$  kümesi içerisinde rassal olarak bir  $k$  ürünü belirle
6:   Do
7:     For Each  $i$  in  $M^V \cup M^{UV}$ 
8:       If  $i \in M^V$  Then
9:          $k$  ürününün  $i$  marketinden satın alınması durumunda  $f'$  hesapla
10:      Else If  $i \in M^{UV}$  Then
11:         $k$  ürününün  $i$  marketinden satın alınması durumunda  $i$  marketinin  $X$  içerisinde yerleştirileceği en iyi pozisyonu belirle ve  $f'$  hesapla
12:      End If
13:       $uc_{ik} = (f' - f) / \min\{q_{ik}, d_k^U\}$ 
14:      Next
15:       $i^* = \operatorname{argmin}_{i \in M^V \cup M^{UV}} \{uc_{ik}\}$ 
16:       $\min\{q_{i^*k}, d_k^U\}$  miktarında  $k$  ürününü  $i^*$  marketinden satın al
17:       $z_{i^*k} = \min\{q_{i^*k}, d_k^U\}$ 
18:       $d_k^U = d_k^U - z_{i^*k}$ 
19:      If  $d_k^U = 0$  Then
20:         $k$  ürününü  $K^{NP}$  kümesinden  $K^P$  kümesine aktar
21:      End If
22:      If  $i^* \in M^{UV}$  Then
23:         $i^*$  marketini  $X$  rotası içerisinde belirlenmiş olan en iyi pozisyona yerleştir
24:         $i^*$  marketini  $M^{UV}$  kümesinden  $M^V$  kümesine aktar
25:      End If
26:    Loop Until  $d_k^U = 0$ 
27:  Loop Until  $K^{NP} = \emptyset$ 

```

Şekil 1. Başlangıç Çözüm Algoritmasına Ait Sözd Kod (Pseudo Code of the Initial Solution Construction Algorithm)

## 5. Sonuçlar (Results)

SZL-GSAP için geliştirilmiş olan iki aşamalı çözüm yönteminin geçerliliği ve etkinliği, Laporte vd. (2003) tarafından tasarlanan ve "<http://jriera.webs.ull.es/TPP.htm>" internet sayfası üzerinden erişim sağlanabilen GSAP problem seti kullanılarak test edilmiştir. Laporte vd. (2003) tarafından geliştirilmiş olan problem seti, kapasiteli/kapasitesiz ve simetrik/asimetrik GSAP problemleri için dört farklı gruba ayrılmaktadır. Her bir sınıfta  $|V|$  ve  $|K|$  değerlerine bağlı olarak farklı büyüklükte problemler yer almaktadır. Problem büyüklüğü lokasyon ve ürün sayısına göre değişmekle birlikte her bir  $(|V|, |K|)$  büyüklüğüne sahip problem seti, talep edilen ürünlerin bulunduğu market sayısını kontrol eden  $\lambda$  parametresine göre çeşitlendirilmiştir. Dikkate alınan üç farklı parametre üzerinden her  $(|V|, |K|, \lambda)$  seti için beş farklı örnek problem üretilmiştir.

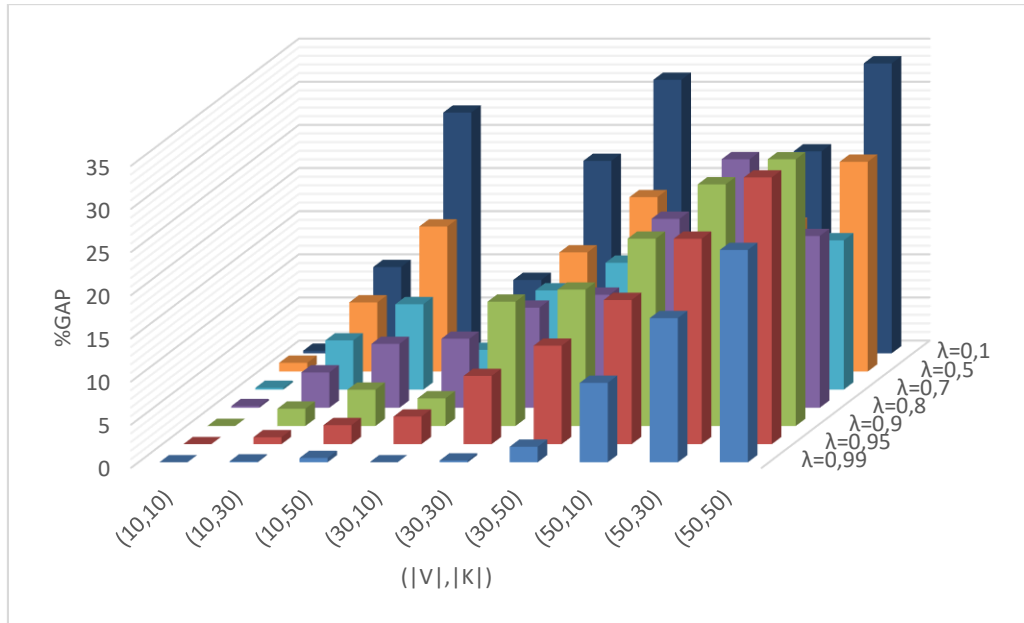
Yapılan bu çalışmada kapasiteli ve simetrik GSAP için geliştirilmiş olan ve büyüklüğü  $(|V|, |K|) = (50, 50)$  olan problemler kullanılmış ve her bir problem; lokasyonlar arası ulaşım süreleri, satın alma süreleri, ürün bozulma ve araç soğutma maliyetleri eklenerek SZL-GSAP için kullanılabilir hale getirilmiştir. Çalışmada, optimum sonucun elde edilebileceği daha küçük boyutlu problemler elde edebilmek için  $|V| = \{10, 30, 50\}$  ve  $|K| = \{10, 30, 50\}$  alınarak problem seti genişletilmiş ve SZL-GSAP için 315 problem üretilmiştir. Problem setinde  $t_{ij} = c_{ij}$ ,  $h_i = 1$ ,  $\theta_1 = \theta_2 = 0.00005$  ve  $\vartheta = 0.01$  olarak varsayılmıştır. Üretilmiş olan problemler için BÇA ile başlangıç çözümü elde edildikten sonra GUROBI 9.1.2 çözücü kullanılarak Intel® Xeon® CPU E5-2643 v3 3.4 GHz işlemci ve 64 GB belleğe sahip bilgisayar üzerinde bir saatlik süre kısıtı ile nihai çözüm elde edilmiştir. Bir saatlik süre sonunda optimum sonuca ulaşamayan problemler için GUROBI ile elde edilen tam sayılı çözümler verilmiştir.

Sayısal uygulamalar sonucunda elde edilen çözümler problem büyüklüklerine göre sınıflandırılarak Tablo 1'de gösterilmiş olup detaylı çözüm sonuçları Tablo Ek 1-3'de verilmiştir. Tablolarda verilmiş olan  $|V|, |K|$  ve  $\lambda$  problem tipini göstermekte olup BÇA ile elde edilen sonuç  $f(BÇA)$ , GUROBI ile elde edilen tam sayılı çözüm  $f(GRB)$ , GUROBI ile elde edilen çözüme ait optimalite aralığı %Gap ve çözüm zamanı CPU ile gösterilmiştir. Tablo 1'de verilmiş olan sonuçlar dikkate alındığında, geliştirilmiş olan çözüm yönteminin özellikle market sayısının ve ürün çeşitliliğinin az olduğu problemler için optimum sonucu veya optimum sonuca yakın çözümler ürettiği tespit edilmiştir. Daha büyük boyutlu problemlerde ise özellikle ürün çeşitliliğinin az olduğu örnekler için elde edilen sonuçlara ait %Gap değerlerinin, ürün çeşitliliğinin fazla olduğu örneklere göre daha kabul edilebilir olduğu görülmüştür.

**Tablo 1.** SZL-GSAP İçin Elde Edilen Sonuçlar (Results for the SZL-GSAP Instances)

V	$\lambda$	K  = 10				K  = 30				K  = 50			
		$f(B\check{C}A)$	$f(GRB)$	%Gap	CPU	$f(B\check{C}A)$	$f(GRB)$	%Gap	CPU	$f(B\check{C}A)$	$f(GRB)$	%Gap	CPU
10	0.1	8884.4	8602.0	0.3	2241.9	27600.6	24244.8	10.0	3600.0	50403.8	35921.2	27.9	3600.0
	0.5	5108.7	4757.5	1.0	3600.0	13430.9	11417.6	8.0	3600.0	22245.2	17335.7	16.8	3600.0
	0.7	3239.4	3095.5	0.2	2229.1	7688.7	6835.9	5.7	3600.0	12246.9	10247.0	9.9	3600.0
	0.8	2577.9	2448.1	0.2	1959.2	5457.5	4772.6	4.1	2880.1	8472.0	6898.6	7.4	3600.0
	0.9	1861.4	1705.4	0.0	53.2	3219.0	2847.9	2.0	3600.0	4462.7	3910.0	4.2	3600.0
	0.95	1458.6	1380.6	0.0	1.4	2525.1	2203.1	0.8	2891.2	3093.5	2769.3	2.2	3600.0
	0.99	1123.3	1106.1	0.0	0.8	1839.2	1558.5	0.1	2163.4	1979.3	1807.9	0.5	3600.0
	30	0.1	11378.4	9837.2	8.5	3600.0	32329.2	24039.4	22.3	3600.0	56372.5	37731.2	31.7
0.5		5979.0	5208.3	5.0	3600.0	14254.4	10901.0	13.8	3600.0	23380.8	15862.6	20.2	3600.0
0.7		3836.3	3340.3	4.6	3600.0	8481.2	6416.9	11.5	3600.0	12422.2	8902.8	14.7	3600.0
0.8		2986.5	2527.1	8.0	3600.0	5855.3	4375.3	11.6	3600.0	7762.1	5855.6	13.1	3600.0
0.9		2296.4	1877.1	3.2	2768.2	3497.6	2687.9	14.4	3600.0	4748.1	3267.4	15.8	3600.0
0.95		1606.3	1388.9	3.2	748.9	2569.7	1879.7	7.9	3600.0	2930.1	2182.4	11.4	3600.0
0.99		1268.7	1025.1	0.0	12.6	1757.2	1495.9	0.2	2635.6	1847.7	1643.6	1.8	3600.0
50		0.1	13994.2	11568.1	12.0	3600.0	39007.1	28192.3	23.4	3600.0	69054.2	44208.1	33.6
	0.5	7412.9	5848.5	14.2	3600.0	17793.1	11817.2	17.0	3600.0	27682.6	16821.5	24.3	3600.0
	0.7	4733.0	3499.2	13.5	3600.0	9502.7	6451.2	17.1	3600.0	13525.8	8576.6	17.3	3600.0
	0.8	3296.3	2454.1	21.9	3600.0	6278.4	4154.9	28.8	3600.0	8450.4	5471.3	19.9	3600.0
	0.9	2123.6	1568.1	21.7	3600.0	3586.6	2364.6	28.0	3600.0	4865.3	2940.8	30.9	3600.0
	0.95	1769.5	1253.8	16.7	2604.5	2607.0	1675.8	23.8	3600.0	3013.7	1892.8	30.9	3600.0
	0.99	1323.2	1132.0	9.2	1561.1	2047.1	1367.8	16.7	3421.4	2248.6	1451.5	24.6	3600.0

SZL-GSAP için elde edilen çözümlere etki eden diğer bir faktör ise problemde ürünlerin var olduğu market sayısını kontrol eden  $\lambda$  parametresidir. Şekil 2'de  $\lambda$  değerine göre elde edilen %Gap değerleri problem büyüklüklerine göre sınıflandırılarak gösterilmiştir.  $(|V|, |K|) = (10, 10)$  olan problemlerde  $\lambda$  değeri değişse de birçok problem için %Gap değeri sıfır veya sıfıra yakın elde edilmiştir.  $(|V|, |K|) = \{(10, 30), (10, 50)\}$  olan problem büyüklüklerinde ise  $\lambda$  değeri azaldıkça sonuçlara ait optimalite aralığının arttığı görülmüştür. Daha büyük boyutlu problemlerde  $\lambda = \{0.8, 0.7\}$  için elde edilen %Gap değerleri,  $\lambda = \{0.99, 0.95\}$  için elde edilen %Gap değerlerine göre kısmen düşük olsa da  $\lambda$  değeri 0.1'e doğru yaklaştıkça elde edilen optimalite aralıkları daha yüksek olmuştur.

**Şekil 2.**  $\lambda$  Değerinin SZL-GSAP Çözümleri Üzerine Olan Etkisi (Effect of the  $\lambda$  on the SZL-GSAP Solutions)

Tablo 1'de verilmiş olan sonuçlara ek olarak Tablo 2'de, GUROBI çözücüsü ile her bir problem büyüklüğüne ait beş örnek için elde edilen optimum sonuç sayıları #Opt ile gösterilmiştir. Problem büyüklüklerine bağlı olarak bu sayılar değerlendirildiğinde problemde  $|V|$  ve  $|K|$  değerlerinin artması optimum sonuca ulaşmayı zorlaştırırken, özellikle  $|K| = 10$  olan problem büyüklükleri için geliştirilmiş olan çözüm yaklaşımının optimum sonuca erişmekte daha başarılı olduğu görülmüştür.



**Tablo 2.** Çözüm Yaklaşımının Performans Analizi (Performance Analyses of the Solution Approach)

V	$\lambda$	K  = 10		K  = 30		K  = 50	
		%Imp	#Opt	%Imp	#Opt	%Imp	#Opt
10	0.1	3.1	2	11.3	0	28.6	0
	0.5	6.7	0	14.6	0	21.8	0
	0.7	4.2	3	10.9	0	16.2	0
	0.8	5.1	3	11.9	0	18.4	0
	0.9	7.0	5	10.5	0	11.7	0
	0.95	4.9	5	11.1	1	10.1	0
	0.99	1.6	5	13.5	3	8.2	0
30	0.1	13.5	0	25.4	0	33.0	0
	0.5	12.7	0	23.2	0	32.1	0
	0.7	12.2	0	23.6	0	28.1	0
	0.8	14.7	0	24.3	0	24.4	0
	0.9	16.9	2	22.4	0	30.2	0
	0.95	12.5	4	25.6	0	24.6	0
	0.99	13.1	5	14.6	2	11.6	0
50	0.1	17.3	0	27.6	0	35.8	0
	0.5	20.9	0	33.5	0	39.2	0
	0.7	25.2	0	31.5	0	36.5	0
	0.8	24.8	0	33.8	0	35.2	0
	0.9	24.5	0	33.6	0	39.1	0
	0.95	26.8	2	35.1	0	36.0	0
	0.99	13.8	3	30.2	1	33.2	0

Tablo 2’de ek olarak GUROBI’den elde edilen nihai sonuçlar ile BÇA sonuçları karşılaştırılmıştır. Bu karşılaştırma için  $f(BÇA)$  ile  $f(GRB)$  arasındaki yüzdelik iyileşme oranı denklem (17) ile verilmiş olan %Imp üzerinden hesaplanmıştır. GUROBI ile elde edilen iyileşme oranları dikkate alındığında özellikle market sayısının ve ürün çeşidi sayısının 30’un üzerinde olduğu problemler için BÇA ile elde edilen çözümler ortalama %25’in üzerinde iyileştirilmiştir. Küçük boyutlu problemlerde ise BÇA’nın optimum sonuçlara yakın sonuçlar ürettiği görülmüştür.

$$\%Imp = \frac{f(BÇA) - f(GRB)}{f(BÇA)} \times 100 \quad (17)$$

## 6. Sonuç ve Tartışma (Result and Discussion)

Yapılan çalışmada literatürde iyi bilinen gezgin satın alıcı problemi, soğuk zincir lojistiğine dayalı olarak genişletilmiş ve bozulabilir ürünler için taşıma kısıtları dikkate alınarak probleme ait yeni bir varyasyon sunulmuştur. Soğuk zincir lojistiğinde gezgin satın alıcı problemi (SZL-GSAP) olarak adlandırılmış olan problem, doğrusal olmayan karışık tamsayı matematiksel model olarak formüle edilmiştir. Problemin çözüm zorluğu nedeniyle çalışmada sezgisel bir yöntem ile entegre edilmiş iki aşamalı bir çözüm yöntemi önerilmiştir. Çözüm yaklaşımının ilk aşamasında çözüm kurucu bir yöntem kullanılarak kaba bir sonuç elde edilirken ikinci aşamada elde edilen bu çözüm GUROBI çözücü ile belirli bir çözüm zamanı sınırı ile geliştirilmektedir. Geliştirilmiş olan çözüm yaklaşımının geçerliliği ve etkinliği, farklı büyüklüklere sahip örnekleri içeren problem seti üzerinde test edilmiştir. Elde edilen sonuçlar, geliştirilmiş olan çözüm yaklaşımının özellikle ürün çeşitliliğinin az olduğu problemler için optimum veya optimuma yakın sonuçlar elde ettiğini göstermiştir. Diğer yandan, ürün çeşitliliği daha fazla olan problemler için bir saatlik zaman kısıtı içerisinde elde edilen sonuçların genel olarak makul bir optimalite aralığına sahip olduğu tespit edilmiştir.

Gerçek hayat uygulamaları dikkate alındığında geliştirilmiş olan çözüm yaklaşımı; özellikle ürün çeşitliliğinin ve market sayısının düşük olduğu uygulamalarda optimum veya optimuma yakın elde edeceği çözümler ile toplam taşıma, ürün satın alma ve ürün bozulma maliyetlerinde önemli tasarruflar sağlayabilecektir. Diğer yandan geliştirilmiş olan matematiksel modelin doğrusal olmayan bir amaç fonksiyonu içeriyor olması büyük boyutlu problemlerde yüksek kalitede çözümlere ulaşmayı zorlaştıran önemli bir faktördür. Operasyonel bir karar alma süreci olarak dikkate alınan bu probleme ait yapılacak gelecek çalışmalar kapsamında daha kısa işlem zamanı ile etkin sonuç üretebilecek meta-sezgisel veya melez bir çözüm yaklaşımı geliştirilebilir. Özellikle literatürde GSAP için geliştirilmiş ve çözüm kalitesi açısından etkinliği kanıtlanmış meta-sezgisel algoritmalar dikkate alınarak algoritmaya ait operatörler SZL-GSAP kısıtlarına dayalı olarak genişletilebilir. Diğer yandan yapılmış olan bu çalışmada dikkate alınan problem, soğuk zincir lojistiğinde ürün alımlarına ve taşımalarına ait daha gerçekçi varsayımlar dikkate alınarak geliştirilebilir. Özellikle bozulma süresi kısa olan ürünler dikkate alındığında problem; ürünlerin üretim zamanları, farklı ürün kalite seviyeleri veya rota sonunda ürünler için beklenen minimum kalite seviyesi kısıtları dahilinde genişletilebilir.

## Çıkar Çatışması (Conflict of Interest)

Yazar tarafından herhangi bir çıkar çatışması beyan edilmemiştir. No conflict of interest was declared by the author.

## Kaynaklar (References)

- Almeida, C.P., Gonçalves, R.A., Goldberg, E.F., Goldberg, M.C., Delgado, M.R., 2012. An Experimental Analysis of Evolutionary Heuristics for the Biobjective Traveling Purchaser Problem. *Annals of Operations Research*, 199(1), 305-341.
- Amorim, P., Parragh, S.N., Sperandio, F., Almada-Lobo, B., 2014. A Rich Vehicle Routing Problem Dealing with Perishable Food: A Case Study. *TOP*, 22, 489-508.
- Angelelli, E., Gendreau, M., Mansini, R., Vindigni, M., 2017. The Traveling Purchaser Problem with Time-Dependent Quantities. *Computers & Operations Research*, 82, 15-26.
- Awad, M., Ndiaye, M., Osman, A., 2021. Vehicle Routing in Cold Food Supply Chain Logistics: A Literature Review. *The International Journal of Logistics Management*, 32(2), 592-617.
- Bianchessi, N., Irnich, S., Tilk, C., 2021. A Branch-Price-and-Cut Algorithm for the Capacitated Multiple Vehicle Traveling Purchaser Problem with Unitary Demand. *Discrete Applied Mathematics*, 288, 152-170.
- Bianchessi, N., Mansini, R., Speranza, M.G., 2014. The Distance Constrained Multiple Vehicle Traveling Purchaser Problem. *European Journal of Operational Research*, 235(1), 73-87.
- Boctor, F.F., Laporte, G., Renaud, J., 2003. Heuristics for the Traveling Purchaser Problem. *Computers & Operations Research*, 30, 491-504.
- Bontoux, B., Feillet, D., 2008. Ant Colony Optimization for the Traveling Purchaser Problem. *Computers & Operations Research*, 35(2), 628-637.
- Cambazard, H., Penz, B., 2012. A Constraint Programming Approach for the Traveling Purchaser Problem. *Principles and Practice of Constraint Programming – CP 2004* (pp. 735-749). Springer.
- Cheaitou, A., Hamdan, S., Larbi, R., Alsyof, I., 2020. Sustainable Traveling Purchaser Problem with Speed Optimization. *International Journal of Sustainable Transportation*, 1-20.
- Chen, H.-K., Hsueh, C.-F., Chang, M.-S. 2009. Production Scheduling and Vehicle Routing with Time Windows for Perishable Food Products. *Computers & Operations Research*, 36, 2311-2319.
- Chen, L., Liu, Y., Langevin, A. 2019. A Multi-Compartment Vehicle Routing Problem in Cold-Chain Distribution. *Computers & Operations Research*, 111, 58-66.
- Choi, M.J., Lee, S.H., 2011. The Multiple Traveling Purchaser Problem for Maximizing System's Reliability with Budget Constraints. *Expert Systems with Applications*, 38(8), 9848-9853.
- Gendreau, M., Manerba, D., Mansini, R., 2016. The Multi-Vehicle Traveling Purchaser Problem with Pairwise Incompatibility Constraints and Unitary Demands: A Branch-and-Price Approach. *European Journal of Operational Research*, 248(1), 59-71.
- Golden, B., Levy, L., Dahl, R., 1981. Two Generalizations of the Traveling Salesman Problem. *Omega*, 9(4), 439-441.
- Gouveia, L., Paia, A., Voß, S., 2011. Models for a Traveling Purchaser Problem with Additional Side Constraints. *Computers & Operations Research*, 38(2), 550-558.
- Hamdan, S., Larbi, R., Cheaitou, A., Alsyof, I., 2017. Green Traveling Purchaser Problem Model: A Bi-Objective Optimization Approach. *7th International Conference on Modeling, Simulation, and Applied Optimization (ICMSAO)*, Sharjah, United Arab Emirates, 4-6 Nisan.
- Hsu, C.-I., Hung, S.-F., Li, H.-C., 2007. Vehicle Routing Problem with Time Windows for Perishable Food Delivery. *Journal of Food Engineering*, 80, 465-475.
- Kang, S., Ouyang, Y., 2011. The Traveling Purchaser Problem with Stochastic Prices: Exact and Approximate Algorithms. *European Journal of Operational Research*, 209(3), 265-272.
- Laporte, G., Riera-Ledesma, J., Salazar-González, J.-J., 2003. A Branch-and-Cut Algorithm for the Undirected Traveling Purchaser Problem. *Operations Research*, 51(6), 940-951.
- Li, Y., Lim, M.K., Tseng, M.-L. 2019. A Green Vehicle Routing Model Based on Modified Particle Swarm Optimization for Cold Chain Logistics. *Industrial Management & Data Systems*, 119(3), 473-494.
- Liu, G., Hu, J., Yang, Y., Xia, S., Lim, M.K. 2020. Vehicle Routing Problem in Cold Chain Logistics: A Joint Distribution Model with Carbon Trading Mechanisms. *Resources, Conservation & Recycling*, 156, 104715.
- Manerba, D., Mansini, R., Riera-Ledesma, J., 2017. The Traveling Purchaser Problem and Its Variations. *European Journal of Operational Research*, 259(1), 1-18.
- Mansini, R., Tocchella, B., 2009. The Traveling Purchaser Problem with Budget Constraint. *Computers & Operations Research*, 36(7), 2263-2274.
- Ong, H.L., 1982. Approximate Algorithms for the Travelling Purchaser Problem. *Operations Research Letters*, 1(5), 201-205.
- Osvald, A., Stirn, L.Z., 2008. A Vehicle Routing Algorithm for the Distribution of Fresh Vegetables and Similar Perishable Food. *Journal of Food Engineering*, 85, 285-295.
- Palomo-Martínez, P.J., Salazar-Aguilar, M.A., 2019. The Bi-objective Traveling Purchaser Problem with Deliveries. *European Journal of Operational Research*, 273(2), 608-622.
- Pearn, W.L. 1991. On the Traveling Purchaser Problem. *Teknik Rapor*.
- Pearn, W.L., Chien, R., 1998. Improved Solutions for the Traveling Purchaser Problem. *Computers & Operations Research*, 25(11), 879-885.
- Qi, C., Hu, L., 2020. Optimization of Vehicle Routing Problem for Emergency Cold Chain Logistics Based on Minimum Loss. *Physical Communication*, 40, 101085.

Riera-Ledesma, J., Salazar-González, J. J., 2005. The Biobjective Travelling Purchaser Problem. *European Journal of Operational Research*, 160(3), 599-613.

Riera-Ledesma, J., Salazar-González, J.-J., 2006. Solving the Asymmetric Traveling Purchaser Problem. *Annals of Operations Research*, 144(1), 83-97.

Riera-Ledesma, J., Salazar-González, J.-J., 2012. Solving School Bus Routing Using the Multiple Vehicle Traveling Purchaser Problem: A Branch-and-Cut Approach. *Computers & Operations Research*, 39(2), 391-404.

Singh, K.N., van Oudheusden, D.L., 1997. A Branch and Bound Algorithm for the Traveling Purchaser Problem. *European Journal of Operational Research*, 97(3), 571-579.

Song, M.-X., Li, J.-Q., Han, Y.-Q., Han, Y.-Y., Liu, L.-L., Sun, Q. 2020. Metaheuristics for Solving the Vehicle Routing Problem with the Time Windows and Energy Consumption in Cold Chain Logistics. *Applied Soft Computing Journal*, 95, 106561.

Teeninga, A., Volgenant, A., 2004. Improved Heuristics for the Traveling Purchaser Problem. *Computers & Operations Research*, 31(1), 139-150.

Voß, S., 1996. Dynamic Tabu Search Strategies for the Traveling Purchaser Problem. *Annals of Operations Research*, 63(2), 253-275.

Wang, Z., Wen, P., 2020. Optimization of a Low-Carbon Two-Echelon Heterogeneous-Fleet Vehicle Routing for Cold Chain Logistics under Mixed Time Window. *Sustainability*, 2020, 1-22.

Zhang, L., Gao, Y., Sun, Y., Fei, T., Wang, Y. 2019. Application on Cold Chain Logistics Routing Optimization Based on Improved Genetic Algorithm. *Automatic Control and Computer Sciences*, 53(2), 169-180.

**Ekler (Appendices)**

**Tablo Ek 1.**  $|V| = 10$  Büyüklüğünde SZL-GSAP Problemleri İçin Elde Edilen Sonuçlar (Results for the SZL-GSAP Instances of Size  $|V| = 10$ )

$\lambda$	Problem No	$ K  = 10$				$ K  = 30$				$ K  = 50$			
		$f(B\check{C}A)$	$f(GRB)$	%Gap	CPU	$f(B\check{C}A)$	$f(GRB)$	%Gap	CPU	$f(B\check{C}A)$	$f(GRB)$	%Gap	CPU
0.1	1	7199.4	6977.7	0.0	34.4	22050.1	21241.5	1.9	3600.0	53256.3	38084.2	28.3	3600.0
	2	9146.6	8780.5	0.1	3600.0	33805.5	27345.4	19.0	3600.0	58718.0	40329.5	30.2	3600.0
	3	10190.2	9698.3	0.0	375.0	25015.3	23889.6	1.3	3600.0	49090.6	35152.0	27.0	3600.0
	4	8287.9	8247.2	0.3	3600.0	27461.6	23310.7	13.6	3600.0	47204.1	33919.9	27.9	3600.0
	5	9598.2	9306.5	1.3	3600.0	29670.5	25436.9	14.0	3600.0	43749.9	32120.3	26.0	3600.0
0.5	1	4200.6	4121.3	0.6	3600.0	10486.5	9351.7	6.8	3600.0	23208.1	18248.4	16.9	3600.0
	2	4929.6	4637.2	1.4	3600.0	16137.4	13766.6	8.1	3600.0	26199.1	19027.2	18.5	3600.0
	3	5729.4	5332.0	0.6	3600.0	12813.3	10605.6	6.7	3600.0	21502.0	17184.7	17.1	3600.0
	4	4787.8	4244.9	0.8	3600.0	12473.3	11252.2	8.0	3600.0	20997.0	16380.6	17.0	3600.0
	5	5896.1	5452.3	1.6	3600.0	15244.0	12111.7	10.6	3600.0	19319.9	15837.6	14.7	3600.0
0.7	1	2698.2	2697.9	0.0	135.5	6239.2	5739.5	4.8	3600.0	12395.6	10618.7	10.0	3600.0
	2	3205.7	3134.7	0.1	3600.0	8829.8	7961.9	6.7	3600.0	14215.7	11470.1	11.5	3600.0
	3	3586.0	3376.2	0.6	3600.0	7040.4	6262.1	4.5	3600.0	11852.0	10196.7	9.6	3600.0
	4	2968.8	2765.7	0.0	209.9	7682.0	6743.8	5.9	3600.0	11200.8	9444.7	9.0	3600.0
	5	3738.2	3502.7	0.2	3600.0	8652.1	7472.0	6.6	3600.0	11570.6	9505.0	9.7	3600.0
0.8	1	2656.4	2543.7	0.3	3600.0	4907.0	4301.2	3.2	3600.0	9343.4	7148.2	7.5	3600.0
	2	2558.1	2487.0	0.0	13.8	5800.8	5487.3	4.0	0.0	9249.7	7655.4	7.7	3600.0
	3	2583.3	2270.1	0.6	3600.0	4326.0	4040.1	3.6	3600.0	8366.3	6718.8	7.8	3600.0
	4	2234.6	2124.3	0.0	130.3	5375.3	4694.5	4.4	3600.0	7505.3	6313.3	7.3	3600.0
	5	2856.8	2815.5	0.0	2451.9	6878.5	5339.6	5.0	3600.0	7895.3	6657.3	6.9	3600.0
0.9	1	1690.4	1623.1	0.0	1.2	2845.3	2540.9	1.8	3600.0	4899.0	4008.2	4.4	3600.0
	2	2095.1	1831.0	0.0	1.5	3960.6	3235.8	2.0	3600.0	4919.0	4424.0	4.4	3600.0
	3	1350.6	1337.9	0.0	196.4	2181.2	2141.1	1.8	3600.0	3519.3	3382.6	3.9	3600.0
	4	1541.0	1500.0	0.0	2.8	3091.8	2832.4	2.0	3600.0	3915.0	3618.4	4.1	3600.0
	5	2629.8	2235.1	0.0	64.0	4016.0	3489.5	2.5	3600.0	5061.4	4116.7	4.0	3600.0

0.95	1	1436.8	1421.7	0.0	1.0	2037.5	1957.7	0.5	3600.0	3055.0	2684.2	2.2	3600.0
	2	1623.0	1269.6	0.0	0.6	2397.0	2338.3	1.2	3600.0	2980.7	2899.8	2.2	3600.0
	3	1323.0	1306.3	0.0	1.3	2868.1	2155.7	1.0	3600.0	3225.1	2859.4	2.1	3600.0
	4	1055.8	1055.8	0.0	0.6	2039.6	1987.7	1.2	3600.0	2672.5	2507.9	2.2	3600.0
	5	1854.6	1849.8	0.0	3.5	3283.4	2576.3	0.0	55.8	3534.1	2895.3	2.1	3600.0
0.99	1	1064.7	1053.0	0.0	0.4	1498.9	1465.0	0.0	7.6	1715.3	1666.3	0.5	3600.0
	2	1145.5	1085.6	0.0	0.4	1601.1	1359.8	0.2	3600.0	1974.0	1689.3	0.6	3600.0
	3	921.4	921.4	0.0	1.2	2302.7	1686.7	0.2	3600.0	2025.3	1995.2	0.4	3600.0
	4	907.3	893.2	0.0	0.4	1478.9	1417.7	0.1	3600.0	1786.6	1670.2	0.5	3600.0
	5	1577.5	1577.5	0.0	1.3	2314.3	1863.3	0.0	9.7	2395.3	2018.6	0.4	3600.0

**Tablo Ek 2.**  $|V| = 30$  Büyüklüğünde SZL-GSAP Problemleri İçin Elde Edilen Sonuçlar (Results for the SZL-GSAP Instances of Size  $|V| = 30$ )

$\lambda$	Problem No	$ K  = 10$				$ K  = 30$				$ K  = 50$			
		$f(B\check{C}A)$	$f(GRB)$	%Gap	CPU	$f(B\check{C}A)$	$f(GRB)$	%Gap	CPU	$f(B\check{C}A)$	$f(GRB)$	%Gap	CPU
0.1	1	10466.3	9093.2	7.5	3600.0	27675.1	21215.7	21.0	3600.0	60772.8	40330.0	32.6	3600.0
	2	11011.5	9442.5	8.8	3600.0	37820.0	27442.1	24.7	3600.0	63566.0	41761.3	33.5	3600.0
	3	11382.3	9904.6	8.9	3600.0	28748.3	21578.6	20.7	3600.0	55268.9	36356.0	30.9	3600.0
	4	11347.5	9839.6	8.3	3600.0	31794.8	24234.2	22.1	3600.0	53264.4	36151.0	31.5	3600.0
	5	12684.5	10906.0	9.1	3600.0	35607.7	25726.6	22.8	3600.0	48990.6	34057.8	29.9	3600.0
0.5	1	5206.4	4853.0	4.2	3600.0	14303.8	9808.1	13.1	3600.0	24394.0	16550.5	21.6	3600.0
	2	5712.5	4998.2	4.2	3600.0	15793.0	12304.6	14.7	3600.0	25682.5	17482.4	21.1	3600.0
	3	6223.8	5263.3	4.3	3600.0	11457.1	9762.6	11.8	3600.0	21311.9	15134.6	20.0	3600.0
	4	5966.4	5106.8	7.3	3600.0	14477.6	11028.5	15.2	3600.0	23948.8	15388.1	19.0	3600.0
	5	6786.0	5820.0	5.0	3600.0	15240.7	11601.4	14.3	3600.0	21566.6	14757.6	19.3	3600.0
0.7	1	3184.6	2958.0	2.3	3600.0	7229.7	5514.2	12.0	3600.0	13265.6	8828.0	18.2	3600.0
	2	3792.7	3310.9	1.9	3600.0	8746.1	6991.4	15.3	3600.0	12994.2	9890.4	14.6	3600.0
	3	3357.5	3155.2	2.6	3600.0	7311.9	5976.9	6.7	3600.0	13046.7	8684.4	12.7	3600.0
	4	3871.0	3256.4	7.5	3600.0	8370.9	6662.4	11.2	3600.0	11670.2	8756.1	12.4	3600.0
	5	4975.6	4021.1	8.8	3600.0	10747.7	6939.6	12.3	3600.0	11134.4	8354.9	15.4	3600.0
0.8	1	3150.1	2543.1	14.3	3600.0	5070.6	4011.2	13.7	3600.0	8518.5	5833.2	16.0	3600.0
	2	2959.7	2423.3	8.3	3600.0	6253.7	4629.1	15.7	3600.0	8054.6	6358.7	15.9	3600.0
	3	2387.8	2254.6	1.6	3600.0	5250.4	4198.3	4.0	3600.0	7443.4	5918.3	8.1	3600.0
	4	2770.7	2438.5	1.7	3600.0	5231.5	4270.3	10.6	3600.0	7120.1	5568.9	11.7	3600.0
	5	3664.0	2975.8	14.2	3600.0	7470.6	4767.4	14.1	3600.0	7674.0	5598.9	13.5	3600.0
0.9	1	2927.0	2167.4	14.2	3600.0	3716.1	2555.5	20.8	3600.0	5347.5	3156.9	24.3	3600.0
	2	2216.3	1909.1	0.0	432.3	3571.4	2731.0	15.5	3600.0	4872.2	3226.7	18.3	3600.0
	3	1622.0	1562.7	0.0	2608.5	2937.9	2765.2	1.8	3600.0	4895.6	3501.0	7.4	3600.0
	4	1949.3	1550.8	0.8	3600.0	3422.8	2579.0	12.3	3600.0	3733.9	3196.3	7.8	3600.0
	5	2767.3	2195.5	0.9	3600.0	3839.6	2808.8	21.6	3600.0	4891.6	3256.0	21.1	3600.0
0.95	1	2537.1	1961.6	16.1	3600.0	2971.4	2166.2	19.7	3600.0	3209.2	2338.7	22.9	3600.0
	2	1022.9	861.1	0.0	22.5	1527.0	1322.4	1.4	3600.0	2056.7	1694.4	2.2	3600.0
	3	1508.2	1280.4	0.0	54.1	2577.7	1942.7	1.1	3600.0	2671.8	2040.9	13.9	3600.0

	4	1367.2	1248.5	0.0	29.2	2511.6	1679.8	0.9	3600.0	2857.8	2193.8	1.2	3600.0
	5	1596.0	1592.7	0.0	38.8	3260.9	2287.4	16.5	3600.0	3854.8	2644.2	17.0	3600.0
0.99	1	2242.7	1490.3	0.0	36.0	1975.3	1607.7	0.3	3600.0	1957.5	1661.3	0.4	3600.0
	2	689.9	682.0	0.0	5.5	911.6	798.9	0.3	3600.0	1079.3	906.3	0.6	3600.0
	3	705.8	698.0	0.0	3.0	1704.9	1595.7	0.0	740.4	1821.7	1664.6	0.3	3600.0
	4	1536.6	1101.3	0.0	11.3	1995.8	1359.9	0.0	1637.4	1971.4	1754.2	0.4	3600.0
	5	1168.6	1153.6	0.0	7.1	2198.6	2117.1	0.3	3600.0	2409.0	2231.7	7.2	3600.0

**Tablo Ek 3.**  $|V| = 50$  Büyüklüğünde SZL-GSAP Problemleri İçin Elde Edilen Sonuçlar (Results for the SZL-GSAP Instances of Size  $|V| = 50$ )

$\lambda$	Problem No	$ K  = 10$				$ K  = 30$				$ K  = 50$			
		$f(B\check{C}A)$	$f(GRB)$	%Gap	CPU	$f(B\check{C}A)$	$f(GRB)$	%Gap	CPU	$f(B\check{C}A)$	$f(GRB)$	%Gap	CPU
0.1	1	13055.3	10789.0	9.1	3600.0	34276.0	25897.1	21.0	3600.0	73566.1	46739.0	34.4	3600.0
	2	13295.2	11345.5	13.1	3600.0	44155.4	31041.8	25.3	3600.0	78772.1	48066.5	36.2	3600.0
	3	14429.5	11751.4	12.1	3600.0	36359.3	26136.0	21.6	3600.0	66437.4	42893.2	33.6	3600.0
	4	13744.9	11291.4	14.3	3600.0	37698.6	27912.1	24.5	3600.0	65113.5	42727.2	32.6	3600.0
	5	15446.1	12663.1	11.2	3600.0	42546.1	29974.4	24.6	3600.0	61381.9	40614.6	31.1	3600.0
0.5	1	6573.5	5737.3	8.0	3600.0	15830.5	11128.4	12.0	3600.0	29859.8	17881.6	21.9	3600.0
	2	7189.5	5562.2	13.9	3600.0	19637.0	12701.4	19.3	3600.0	29255.5	18234.6	24.4	3600.0
	3	7820.0	6148.7	11.4	3600.0	16692.3	10989.6	17.2	3600.0	26866.2	16321.2	28.7	3600.0
	4	7509.5	5615.7	15.1	3600.0	17702.4	11688.0	21.1	3600.0	27192.9	15825.9	27.7	3600.0
	5	7971.8	6178.7	22.5	3600.0	19103.4	12578.7	15.3	3600.0	25238.5	15844.1	18.6	3600.0
0.7	1	3805.7	3347.7	2.7	3600.0	8165.9	6111.1	16.1	3600.0	14058.2	8717.6	17.4	3600.0
	2	4846.5	3405.3	12.6	3600.0	9883.2	6803.1	23.8	3600.0	14594.0	8950.3	19.7	3600.0
	3	4494.9	3612.8	16.7	3600.0	8780.8	6197.0	12.5	3600.0	13103.8	8636.9	15.4	3600.0
	4	5074.4	3450.7	8.1	3600.0	9156.2	6274.4	18.1	3600.0	12740.1	8155.8	20.4	3600.0
	5	5443.4	3679.5	27.6	3600.0	11527.5	6870.2	14.9	3600.0	13132.8	8422.2	13.8	3600.0
0.8	1	3534.9	2472.3	28.9	3600.0	5960.1	4021.5	25.5	3600.0	8916.8	5704.6	23.0	3600.0
	2	2830.6	2465.0	12.9	3600.0	6419.2	4238.2	33.0	3600.0	8294.2	5750.7	16.0	3600.0
	3	3527.4	2427.7	24.7	3600.0	6015.3	3979.3	23.2	3600.0	8269.6	5537.3	12.7	3600.0
	4	3000.4	2504.0	11.0	3600.0	6094.0	4116.3	26.8	3600.0	8082.9	5050.6	22.2	3600.0
	5	3588.2	2401.3	32.2	3600.0	6903.3	4419.0	35.6	3600.0	8688.8	5313.3	25.5	3600.0
0.9	1	2571.8	1708.3	33.5	3600.0	3817.8	2373.4	23.5	3600.0	4826.1	2985.0	35.8	3600.0
	2	2282.1	1543.0	25.9	3600.0	3940.1	2429.7	32.6	3600.0	4739.2	3047.4	33.4	3600.0
	3	1857.0	1501.0	17.2	3600.0	3141.6	2363.5	22.1	3600.0	5692.4	2987.8	24.7	3600.0
	4	1679.8	1627.8	0.2	3600.0	3297.1	2275.8	26.0	3600.0	4240.8	2866.8	20.8	3600.0
	5	2227.5	1460.7	31.8	3600.0	3736.2	2380.4	35.9	3600.0	4827.8	2817.0	39.8	3600.0
0.95	1	2092.9	1374.6	30.5	3600.0	2850.4	1744.0	37.7	3600.0	3403.5	1884.5	43.0	3600.0
	2	1077.9	1060.1	0.0	73.8	2124.0	1568.4	1.0	3600.0	2184.3	1756.7	2.5	3600.0
	3	1940.2	1375.3	0.0	2148.6	2613.4	1618.7	26.0	3600.0	2995.1	1830.6	33.7	3600.0
	4	1940.7	1369.5	14.9	3600.0	2911.2	1706.8	25.4	3600.0	3326.5	1968.8	40.5	3600.0
	5	1795.9	1089.5	38.1	3600.0	2536.0	1741.3	28.8	3600.0	3159.1	2023.6	35.1	3600.0

---

0.99	1	1556.1	1450.9	0.0	212.2	1855.9	1617.7	0.1	3600.0	2035.7	1704.3	0.5	3600.0
	2	974.7	948.6	0.0	48.4	1369.4	1249.3	0.0	2706.9	1503.1	1221.3	7.0	3600.0
	3	1238.2	1119.5	0.0	344.9	2484.6	1328.0	27.7	3600.0	2489.5	1402.2	33.8	3600.0
	4	1478.8	1185.0	19.1	3600.0	2481.0	1403.2	26.5	3600.0	2784.8	1531.7	40.0	3600.0
	5	1368.3	956.2	26.8	3600.0	2044.5	1241.0	29.2	3600.0	2430.1	1397.8	41.6	3600.0

---