



YÖNETİM BİLİŞİM SİSTEMLERİ DERGİSİ
<http://dergipark.ulakbim.gov.tr/ybs>

Yayın Geliş Tarihi: 23.11.2017
Yayına Kabul Tarihi: 30.11.2017
Online Yayın Tarihi: 20.12.2017

Cilt:3, Sayı:2, Yıl:2017, Sayfa:105-113
ISSN: 2148-3752

MULTI-VEHICLE DYNAMIC ROUTING WITH TIME-WINDOWS IN PARCEL DELIVERY

Nafiz ÜNLÜ

Istanbul Technical University, Turkey

Erdem UÇAR, Gökay Burak AKKUŞ*

Trakya University, Turkey

Buğra ŞEN

Abstract: CEP companies perform pickup and delivery with limited warehouse usage daily. Customers expect real time monitoring, flexible pick-up and delivery models, and CEP companies are working on technology based solutions to meet such requirements. In this paper, we present a method to solve VRPTW (vehicle routing problems with time windows), which is an NP-hard problem. The method used is divide and conquer where a dynamic routing mechanism is devised to optimize the work load within a Courier Working Zone to meet possible changes in the task list during the working period.

Keywords: Pickup and delivery problem, Vehicle Routing Problem, Multi-vehicle Dynamic Routing with Time Windows.

KARGO İŞLEMEDE ZAMAN PENCERELİ ÇOK ARAÇLI DİNAMİK ROTALAMA

Özet: Kargo firmaları kısıtlı depo kullanımı ile günlük olarak toplama ve dağıtım işlemleri gerçekleştirmektedir. Müşteriler gerçek zamanlı izleme ve esnek dağıtım toplama modellerini kargo firmalarından talep etmekte ve kargo firmaları da bu taleplere teknolojiye dayanan çözümlerle cevap vermeye çalışmaktadır. Bu çalışmada böl ve fethet yöntemleri kullanılarak belli bir kurye çalışma alanında gün içi değişikliklere de cevap verebilecek dinamik bir rotalama oluşturulması ve iş yükünün en iyilenmesi hedeflenmektedir. Tek araçlı, pencereci dağıtım ve toplama problemi kargo sektörü için önemli bir konu olarak karşımıza çıkmaktadır. Bu çalışmada bu probleme yaklaşımlar geliştirilmektedir.

Anahtar Kelimeler: Toplama ve Dağıtım Problemi, Araç Rotalama Problemi, Pencereci Çok Araçlı Dinamik Rotalama.

*Contact Author: gokayakkus@trakya.edu.tr, Trakya Üniversitesi, Edirne, Türkiye

GİRİŞ

Mevcut kargo firmalarının gündemini verimlilik başlıklı projeler oluşturmaktadır. Bu projeler yoğunlukla araç doluluk oranlarının artırılması, yakıt tüketiminin azaltılması, sürüş sürelerinin azaltılması, trafikte geçirilen sürenin azaltılması, iletişim sorunlarından kaynaklanan tekrar aynı adreslere gitmenin engellenmesi, ilk seferde teslimat oranlarının artırılması, koli ve dosya başına elleçleme sayılarının düşürülmesi gibi sıralanabilecek konuları içermektedir. Bu konularda gerçekleştirilecek iyileşmeler hem ticari verimliliği getirecek hem de çevresel kirlilik gibi konularda duyarlı bir yaklaşım gösterilmiş olmasını sağlayacaktır.

Bunlara ek olarak özellikle e-ticaret tarafındaki gelişme trendi taşıma ihtiyaçlarını da büyüme yönlü etkilemektedir. E-ticaretin payı ülkemizde perakende pazar için %2 olarak görülmekte, gelişmiş ülkelerde ise bu ortalama %7.1 dolayında bulunmaktadır. Bu da önümüzdeki dönemde büyüme eğiliminin net ve güçlü şekilde yukarı yönlü olacağını göstermektedir.

Artacak olan ticaret, kargo firmalarına da kapasite büyütme ihtiyacı olarak geri dönecektir. Bu ek kapasite ihtiyacı tek başına personel sayılarındaki artışlarla çözülemeyeceği gibi daha uzun vadede yapılacak transfer merkezi veya ayırıştırma makine otomasyonları ile de desteklenmesi şart olmakla birlikte verimli şekilde çözülemeyecektir. Çözüm, verimliliği sağlayacak otomasyona dayalı en iyilemeden geçmektedir.

En iyileme için ilk ele alınan problem kuryelerin kendilerine düşen yükü hangi sırada dağıtmasının daha verimli olacağını bulunmasıdır. Bu da sonuçta araç rotalarının etkin şekilde belirlenmesiyle sağlanabilecektir.

Araç Rotalama Problemi (ARP) belli bir yükün belli noktalardan yola çıkan araçlarla müşterilere ulaştırılması problemini tarif etmektedir [1]. Kargo firmaları kendilerine ulaşan kolileri veya dosyaları Transfer Merkezlerinde işleyerek ayırıştırırlar ve daha önceden belirlenmiş alanlar bazında kümelerler. Her bir küme bir alanı ifade eder ve bu alana Kurye Çalışma Alanı adı verilir. Kurye Çalışma Alanı (KÇA) mevcut kargo işleyişlerinde genellikle bir şubenin her bir kuryesinin kişisel sorumluluğunda olan alt alanlarına verilen isimdir. Bu çalışmada ilgili alanların dinamik olarak uygulama tarafında belirleneceği ve en iyileme sayesinde verimlilik oluşturulacağı düşünülmektedir.

Araç Rotalama Problemlerinin tanımlanması tarih olarak 1959 yılına kadar gitmektedir [3]. Klasik Araç Rotalama Problemine eklenen kısıtlarla ARP konusunda yeni ve daha detaylı problemler türetilmektedir. Zaman Pencere Araç Rotalama Problemi; araç rotalama problemlerindeki her bir noktanın üzerine servis sunulacak [e,1] zaman aralığı kısıtının eklenmesi ile geliştirilen bir türdür..

YÖNTEM

Bu çalışma kapsamında “Rota”, belli bir aracın (kuryenin) belli bir noktadan başlayarak ardışık olarak ziyaret edeceği noktaların teslimat veya dağıtıma bağlı müşteri tercihlerini gözetecek ve ayrıca araç, trafik, kurye vardiyaları gibi etmenlerden gelecek kısıtları da sağlayacak şekilde arka arkaya sıralanmasını tanımlamaktadır.

Rotalama probleminin çözümünde gözetilmesi gereken bir başka nokta da, özellikle gerçek hayatta müşterilerin katma değer olarak gördüğü belli bir zaman dilimi içinde servis alma talebinin karşılanmaya çalışılmasıdır. Bu kapsamda problem tanımında bir kuryenin ziyaret edeceği her bir nokta için bir zaman dilimi ataması yapıldığı varsayılmıştır [2].

Bir kurye tarafından ziyaret edilecek noktaları $N = \{n_0, n_1, n_2, \dots, n_m\}$ kümesi temsil etmektedir. n_0 ise başlangıç noktasını göstermektedir ve her bir n_i , $i = 1, 2, \dots, m$ müşteri teslimat ya da toplama adresini ifade etmektedir. Her bir $\langle n_i, n_j \rangle$ ikilisi bir noktadan (n_i) diğerine (n_j) olan ulaşımı, diğer bir değişle sürüşü ifade etmektedir. Bu ikili gösterimden yola çıkarak sürüş süresini de tekli ifadeyle t_{ij} olarak tanımlayabiliriz. Benzer şekilde d_{ij} iki nokta (n_i ile n_j) arasındaki uzaklığı ifade etmektedir. Veri kümesinde yer alacak her bir noktanın bir diğer noktaya uzaklığı da d_{ij} elemanlarından oluşan bir matrisle (D matrisi) ifade edilmektedir. Problemin çözülmesinde basitliği sağlamak açısından yolların tek-yön ya da gidiş-geliş olabileceği veya trafik durumları gözetilmemiş, bunun sonucu olarak da $t_{ij} = t_{ji}$ olduğu varsayılmıştır.

Zaman Pencereci Dinamik Rotalama

Zaman pencereci yaklaşım nedeniyle her bir n_i için bir servis zaman aralığı bulunmaktadır. Bir kuryenin performansını belirleyen temel unsurlardan biri de müşteri tarafından belirlenmiş olan e_i ve l_i saatleri arasında ($e_i < l_i$) ilgili noktanın (n_i) ziyaret edilerek servis sunulmasıdır. Bu kısıt en iyileme aşamasında geliştirilecek çözüm için önemli bir girdi olarak kullanılacaktır. Bunlara ek olarak s_i , ilgili noktaya (n_i) ulaşıldığında kuryenin yaptığı göreve ilişki orada ne kadar bir süre harcayacağını tanımlamaktadır. Bu süre kurye görevinin büyüklüğü veya alınacak koli veya dosyalara ait bilgilerin daha önceden sistemler arasında ne kadar işlenmiş olduğuyula ilişkilidir. e_i olarak tanımlı kısıtların bulunması, aynı zamanda bir noktaya ulaşıldığında servis verilmesi konusunda eğer ilgili saatten önceki bir zaman diliminde bulunuluyorsa, belli bir süre beklenmesini de gerektirebilir.

Zaman Pencereci Araç Rotalama Problemi (ZPARP) NP-zor problemlerdendir, çözüm için yaklaşımlar geliştirilmiştir [4]. ZPARP için banka taşımaları, posta-kargo taşımaları, atık taşımaları, okul servis planlaması problemleri örnek verilebilir [5].

Problemin çözümü olarak sıralı şekilde $v_k, k = 1, 2, \dots, m-1$ olmak üzere bir vektör elde edilmiş olması beklenmektedir. Çözüm vektörünün v_0 başlangıç elemanı n_0 noktasından n_1 noktasına t_{01} süresinde gidilerek $[e_1, l_1]$ zaman penceresi içinde kalarak s_1 süresi kadar bir toplama veya dağıtım servisi verildiğinde başarılı olarak görülecektir.

BULGULAR

Parçala fethet yaklaşımının uygulanması; Her bir zaman penceresi kümesi için minimum gerekli araçların belirlenmesi ve bu belirlenen gruplar için ayrı ayrı küçük çözümler üretildikten sonra farklı zaman pencereleri için çözümlerin birleştirilebilir, birleştirilemeyeceğinin belirlenerek gerekli olan minimum araç sayısı ve sonuçta ulaşılan her çözümün rotalanması olarak uygulanacaktır. Bu çalışmada, her bir zaman penceresi grubunun dağıtım toplama modelinin çözümü için sezgisel veya yapay sinir ağları algoritmaları uyarlanıp karşılaştırılmaktadır.

Her bir zaman penceresi Kümesi $ZP_0 = \{n_i, n_{i+1}, n_{i+2}, \dots, n_{i+n}\}$ şeklinde bir küme oluştursun, Oluşturulan Rota için gerekli olan toplam süre $T = v_k$ Vektöründeki tüm t_{ij} süreleri ve s_j toplama veya dağıtım için geçen sürelerin toplam değeri ifade edelim.

$$ZP_0 = \{n_a, n_{a+1}, n_{a+2}, \dots, n_{a+n}\}, T_0$$

$$ZP_1 = \{n_b, n_{b+1}, n_{b+2}, \dots, n_{b+n}\}, T_1$$

$$ZP_2 = \{n_c, n_{c+1}, n_{c+2}, \dots, n_{c+n}\}, T_2$$

$$ZP_n = \{n_n, n_{n+1}, n_{n+2}, \dots, n_{n+n}\}, T_n$$

$$ZP_n = \{n_n, n_{n+1}, n_{n+2}, \dots, n_{n+n}\}, T_n$$

Oluşan her vektör bu zaman pencerelerine uyum açısından değerlendirilmiş, bir sonraki iterasyon için bu uyum değeri girdi olarak kullanılmıştır.

Önerilen çözümün geliştirilmesi Knime aracı üzerinde gerçekleştirilmiştir.

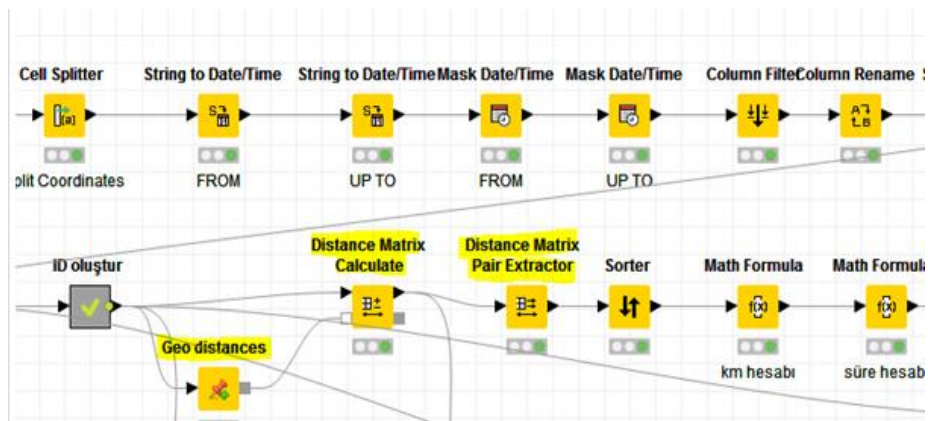
Veri yapısı aşağıdaki şekilde (Şekil-1) bir kargo firmasından anonim özelliklerde sağlanmıştır.

Şekil - 1

S ADRES	S X,Y	I Süre(sn)	S FROM	S UP TO
Acbadem, Acbadem Cd., 34718 Kadıköy/İstanbul	40.994980, 29.033680	7	1899-12-31T06:00:00	1899-12-31T08:00:00
İsmail Hakkı Bey Sk. 21a, Acbadem, 34718 Kadıköy/İstanbul	40.995574, 29.032893	7	1899-12-31T06:00:00	1899-12-31T08:00:00
Bedrettin Sokağı No:14, Acbadem, 34718 Kadıköy/İstanbul	40.996747, 29.033345	7	1899-12-31T06:00:00	1899-12-31T08:00:00
Taşköprü Cd. No:28, Acbadem, 34718 Kadıköy/İstanbul	40.994090, 29.033295	7	1899-12-31T06:00:00	1899-12-31T08:00:00
Nazıfbey Sk. No:3, Acbadem, 34718 Kadıköy/İstanbul	40.993693, 29.034625	7	1899-12-31T06:00:00	1899-12-31T08:00:00
Hacı Recep Sk. No:19, Hasanpaşa Mahallesi, 34718 Kadıköy/İstanbul	40.993272, 29.035333	7	1899-12-31T06:00:00	1899-12-31T08:00:00
Sarayardı Cd. No:66, Acbadem, 34718 Kadıköy/İstanbul	40.996122, 29.036073	7	1899-12-31T06:00:00	1899-12-31T08:00:00
Şamfıstığı Sokağı No:2, Acbadem, 34718 Kadıköy/İstanbul	40.993806, 29.034614	7	1899-12-31T06:00:00	1899-12-31T08:00:00
Uzun Hafız Sk. No:7, Rasimpaşa Mahallesi, 34718 Kadıköy/İstanbul	40.994928, 29.025110	7	1899-12-31T08:00:00	1899-12-31T13:00:00
Ayşekadin Aşevi, Rasimpaşa Mahallesi, Teyyare Mahallesi, 34718 Kadıköy/İstanbul	40.993760, 29.025572	7	1899-12-31T08:00:00	1899-12-31T13:00:00
Mühürdar Fuat Sk. No:18, Osmanağa Mahallesi, 34718 Kadıköy/İstanbul	40.991930, 29.026535	7	1899-12-31T08:00:00	1899-12-31T13:00:00
Mühürdar Baş Sk. No:12, Caferağa Mahallesi, 34718 Kadıköy/İstanbul	40.988078, 29.021643	7	1899-12-31T08:00:00	1899-12-31T13:00:00
Sivastopol Sk. No:15, Caferağa Mahallesi, 34718 Kadıköy/İstanbul	40.986500, 29.023862	7	1899-12-31T08:00:00	1899-12-31T13:00:00
Murat Bey Sokağı No:17, Caferağa Mahallesi, 34718 Kadıköy/İstanbul	40.985937, 29.024361	7	1899-12-31T08:00:00	1899-12-31T13:00:00
Alageyik Sk. No:6, Fenerbahçe Mahallesi, 34726 Kadıköy/İstanbul	40.975398, 29.047889	5	1899-12-31T08:00:00	1899-12-31T16:00:00

Knime aracılığı ile X,Y Kolonunda bulunan string koordinat verileri Latitude,longitude haline getirilip, her bir adres verisi için bir ID atandı. Oluşturulan ID'ler arasındaki uzaklık matrisi Knime üzerinde bulunan "Geo Distances, Distance Matrix Calculate,Distance Matrix Pair Extractor Node" ları sayesinde (Şekil-2) aşağıdaki veri yapısına (Şekil-3) getirildi.

Şekil - 2



Oluşturulan veri R ortamında kaynak olarak kullanılarak, Tsp Solver kütüphanesi ile işlendi. Algoritmaların beklediği veri biçimine getirmek için (50 nokta üzerinden, 50x50 lik bir matris ve her bir hücrede ise ilgili kesişimdeki uzaklık bilgisi tutulmaktadır).

Şekil - 3

Object1	Object2	Uzaklık (KM)	Sure (dk)
5	8	0,020275959	0,024331
13	14	0,121219373	0,145463
5	6	0,121740632	0,146089
6	8	0,136245897	0,163495
48	49	0,145765222	0,174918
36	37	0,148701075	0,178441
1	2	0,150326288	0,180392
33	34	0,15652453	0,187829
1	4	0,167540562	0,201049
14	36	0,171375238	0,20565
7	30	0,181658126	0,21799
15	42	0,183284148	0,219941
4	8	0,185262354	0,222315
4	5	0,193178905	0,231815
32	35	0,194156198	0,232987
26	27	0,195458497	0,23455
28	29	0,200562834	0,240675
25	26	0,203906475	0,244688

Oluşan matrise ait özet görüntü (Şekil-4) aşağıdaki gibidir:

Şekil - 4

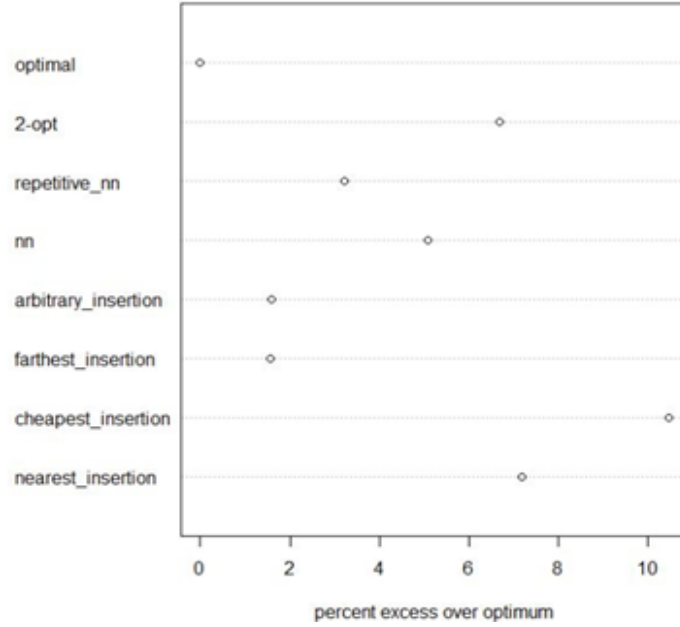
```
> mesafe1
      [,1]      [,2]      [,3]      [,4]      [,5]      [,6]      [,7]      [,8]
[1,] 0.0000000 0.1503263 0.3194270 0.1675406 0.26331402 0.3785090 0.3823989 0.24505446
[2,] 0.1503263 0.0000000 0.2186068 0.2710570 0.40991468 0.5275522 0.4405579 0.39259723
[3,] 0.3194270 0.2186068 0.0000000 0.4755205 0.57320911 0.6773487 0.3850538 0.55350098
[4,] 0.1675406 0.2710570 0.4755205 0.0000000 0.19317891 0.3117715 0.5225040 0.18526235
[5,] 0.2633140 0.4099147 0.5732091 0.1931789 0.0000000 0.1217406 0.4766440 0.02027596
[6,] 0.3785090 0.5275522 0.6773487 0.3117715 0.12174063 0.0000000 0.5197116 0.13624590
[7,] 0.3823989 0.4405579 0.3850538 0.5225040 0.47664401 0.5197116 0.0000000 0.45891429
[8,] 0.2450545 0.3925972 0.5535010 0.1852624 0.02027596 0.1362459 0.4589143 0.00000000
[9,] 1.1575555 1.0575535 1.1589095 1.1156495 1.30403237 1.4122426 1.4960555 1.29928896
[10,] 1.1166768 1.0407450 1.1781069 1.0448035 1.22283666 1.3212998 1.4799743 1.22131882
[11,] 1.1087196 1.0782910 1.2605973 0.9915213 1.13735443 1.2123787 1.4907748 1.14170769
[12,] 2.0418120 2.0269393 2.2147071 1.9064555 2.02102444 2.0696313 2.4230272 2.02986999
```

Verilen metotlara (2-opt, repetitive-nn, arbitrary_insertion, farthest_insertion, cheapest_insertion, nearest_insertion) göre hesaplanan turlar ve toplam mesafe bilgileri ve sonuçların karşılaştırılması aşağıda gösterilmektedir (Şekil-5);

(Optimal olarak 30 km verilmiştir*)

\$nearest_insertion, \$cheapest_insertion, \$farthest_insertion ve \$arbitrary_insertion metotları kullanılmıştır.

Şekil - 5



Verilen metotlara göre oluşturulan Rota Vektörleri aşağıda listelenmektedir:

```
> as.integer(tours[[1]])
34 11 39 40 15 16 18 19 20 24 25 26 27 45 49 50 48 47 46 44 43 28 29 23
22 21 17 42 41 38 36 37 14 13 12 10 9 35 31 32 3 30 7 2 1 8 6 5 4 33
```

```
> as.integer(tours[[2]])
28 29 49 50 48 47 46 45 44 43 26 23 19 18 15 42 39 38 36 14 13 12 9 10
34 35 31 32 3 30 7 2 1 8 6 5 4 33 11 37 40 41 16 17 20 21 22 24 25 27
```

```
> as.integer(tours[[3]])
6 50 49 48 47 46 45 44 43 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15
42 41 40 39 38 37 36 14 13 12 11 10 9 34 33 32 35 31 2 3 30 7 1 4 8 5
```

Yukarıda elde edilen vektörler, kurye alan bilgisine sahip deneyimli bir kuryenin olası çizimiyle kıyaslanmıştır. Şekil-5'te ilk satırda bulunan "opt" değeri

bu çizimi ifade etmektedir. Genel olarak karşılaştırma ve ticari olarak geçerlilik, bu çözüm vektörü temel alınarak gerçekleştirilmiştir.

TARTIŞMA ve SONUÇ

Kargo firmaları için şubeli veya şubesiz modelde kuryelerin belli bir alanda vardiyalarına göre ayrıştırılmış olan dosya-koli toplama veya dağıtım görevlerini en verimli şekilde yerine getirmesi günümüzde en önde gelen problemlerdendir.

Bu çalışmada önerilen yöntemle gerçek hayat kısıtları çok fazla basitleştirilmeden problem alanında bir modelleme yapılmış, yeni gelişen gerçek zamanlı müşteri taleplerini destekleyecek dinamik bir rotalama sağlanmıştır. Özellikle zaman pencereli kısıtın eklenmesi çözümün tasarlanmasında önemli bir karmaşıklık getirmektedir.

Ticari faaliyete yeni başlamakta olan bir kargo firmasıyla sonuçlar değerlendirildiğinde ticari olarak kullanılabilir bir yöntem oluşturulduğu karşılıklı mutabakatla görülmüş, sonraki çalışmalarda operasyonel uygulamalara doğrudan ilgili yöntemin yazılımın parçası olarak eklenmesine ve uygulamadaki verimliliğin ölçülmesine karar verilmiştir.

İleriye dönük çalışmalarımız çoklu değişkenlere dayalı en iyileme alanında başka yöntemlerin de denenmesini içermektedir.

KAYNAKLAR

[1] M. I. Hosny, C. L. Mumford, The single vehicle pickup and delivery problem with time windows: intelligent operators for heuristic and metaheuristic algorithms. *Journal Heuristics*, 16, 417–39 (2010).

[2] Lu, Q. and Dessouky, M. M. (2006). A new insertion-based construction heuristic for solving the pickup and delivery problem with time windows. *European Journal of Operational Research*, 175(2):672–687.

[3] Dantzig, G.B., Ramser, J.H.,1959, “The truck dispatching problem”, *Management Science* , 6-80.

[4] Savelsbergh M.W.P., 1985, “Local search in routing problems with time windows”, *Annals of Operations Research*, 4, 285-305.

[5] Solomon M.M., Desrosier J.,1988, “Survey Paper: The time constrained routing and scheduling problems”, *Transportation Science*, 22, 1.