nuçlara ulaşılmasına imkan verebilecektir.

4. SONUÇ

Sayısal ya da sayısallaştırılmış haritası mevcut bir yağış alanı için, gerekli bölgesel parametrelerin ve bilgilerin olması durumunda, bir havzanın ya da yağış alanının bir çıkış noktası için, yağış şiddetine yağışın başlangıç zamanına ve süresine bağlı olarak bir taşkın riskinin olup olmayacağının, olacaksa yağış başlangıcından ne kadar zaman sonra olabileceğinin; o noktadaki hidrografın elde edilmesi, burda önerilen yaklaşımın uygun yazılım ve donanım imkanları ile desteklenmesi halinde, kestirilmesi, taşkının muhtemel etki alanı sınırlarının belirlenmesi, anında mümkün olabilecektir.

Ulaşılacak sonucu güvenirliği.işleme giren bilgilerin doğruluğuna, öngörülen isabetine bağlı olacaktır. Bu arada oluşturulan sistemin.zaman içinde yapılacak gözlemlerden sağlanacak bilgilerle test edilerek, havza elemanı boyutlarını, nitelik farklarına göre küçülterek.daha doğru güvenilir sonuçlara ulaşılmasıda mümkün olacaktır. Ancak burda anahtar bilgiler olan yağış şiddeti ve başlangıç zamanı, süresi; yağışın işlemekonu havzanın tamamınadüşüp düşmediği v.b hususların doğru ve güvenilir olarak belirlenmesindeki güçlükler ile, bu bilgilerin anında bilgi-işlem merkezine ulaştırılası, risk bölgelerine duyurma, bu konuda uygun yazılım sağlanması, sistemin oluşturulmasındaki ana güçlükleri oluşturacaktır. Bu nedenle, özellikle, genişhavzalar için sistemden sağlanacak sonucun güvenilirliği .ve başarısı, daha çok, bu güçlüklerin aşılmasına bağlı olacağı gözönünde bulundurulmalıdır.

Ülkemiz ve özellikle de, yeryeşim alanlarının dere ve ırmak ağızlarında yer aldığı.yağışlarm bol, arazinin fazla eğimli olduğu Karadeniz Bölgesi yerleşim alanları için daha da önemli olduğuna inandığımız bu kapsamdaki proje ve araştırmaların mutlaka yapılmasında, ilgili kuruluşlarca desteklenmesinde büyük yararlar görmekteyiz.

KAYNAKLAR

Beyazıt, M., (1987) $(1987) "Hidroloji" İstanbul Teknik Üniversitesi Matbaası İSTANBUL

Kratzch, H. ,: (1983) Mining Subsidence Engineding, Spring Verlag, Berlin.

Kızılkaya, T. , Yer gül, Ü. ,: 1981 Su Yapıları

Kuşçu, Ş. , ve Arkadaşları: Bursa Metropoltan Alanı İçin Sayısal Harita Tabanlı Taşkın Tahmin ve Erken Uyarı Sistemi Oluşturma proje Önerisi 04. 08. 1995, ZKÜ. ZONGULDAK

Önal. , M.R. , ve Arkadaşları: (19937 Küçük Havzalarda Yüzey Akım Miktarının Bulunmasında Kullanabilecek Parametrelerin tayini . , Türkiye Ulusal Jodezi-Jeofizik Birliği (TUJJB) Genel Kurul Bildiri Kitabı, ANKARA

Yanmaz,A.M. Coşkun. ,F., (1994)Küçük Havzalarda Tasarım Hidrografın Bulunması Üzerine Bir Çalışma. , TUJJB Genel Kurulu Bildin Kitabı ANKARA (545)

büyük jeodezik ağların dengelenmesinde karşılaşılan güçlükler we çözüm önerileri

Doç. Dr. Sebahaîtin Bektaş

ÖZET: Bilgisayarların sunduğu geniş ve hızlı hesaplama olanaklarına rağmen büyük jeodezik ağların dengelenmesi özellikle PC bazlı bilgisayarlarda sorun olmakta, bellek yetmezliği ortaya çıkmakta ve hesaplama süresi çok uzamaktadır. Bilinmeyen sayısının karesi ile orantılı olarak artan bellek gereksinimi, bilinmeyen sayısının küpüyle orantılı olarak artan işlem (hesaplama) süresi özellikle bellek kapasitesi ve hızı yetersiz bilgisayarlarda sorun yaratmaktadır. Bu çalışmada konu ele alınmış mevcut bilgisayarla maximum büyüklükte jeodezik ağların (Nivelman, Gravite, Nirengi, Üç Boyutlu) dengelenebilmesi için alışılagelmiş dengeleme algoritmasında bir takım değişiklikler yaparak bellek gereksinimini ve hesaplama süresini en aza indiren bir çözüm yöntemi sunulmuştur.

GİRİŞ: Bilindiği gibi jeodezik ağlarda boyut sayısı arttıkça bilinmeyen sayısı da artmaktadır. Tek boyutlu nivelman ya da gravite ağlarında nokta sayısı kadar *(dh), (dg)* kadar, iki boyutlu nirengi ağlarında nokta sayısının iki katı kadar *(dx, dy)* üç boyutlu jeodezik ağlarda nokta sayısının üç katı kadar *(dx, dy, dz)* bilinmeyen sayısı olmakta noktalardaki çekül sapmasının yönü de belirlenmek istendiğinde bilinmeyen sayısı nokta sayısının beş katına *(dx, dy, dz, dç, dX)* her nokta için ayrı refraksiyon katsayısının belirlenmesi istendiğinde bilinmeyen sayısı nokta sayısının altı katına *(dx, dy, dz, dç, dX,dk)* kadar çıkabilmektedir. Yatay doğrultuların gözlendiği iki ve daha çok boyutlu ağlarda yatay doğrultu gözlemi yapılan her nokta başına bir de yöneltme bilinmeyeni geldiğini unutmamak gerekir.

Dengeleme İşleminin Yapılması

Çeşitli dengeleme yöntemleri olmakla beraber *endirekt (dolaylı) ölçüler dengelemesi* bilgisayar programlamaya uygunluğu ve sadeliği nedeniyle tercih edilmektedir. Kaba hatalı ve uyuşumsuz ölçülerden arındırılan gözlemlerin dengelenmesi için bugün de geçerliliğini koruyan *Gauss'un en küçük kareler ilkesi'ne* göre dengeleme yapılmaktadır. Ağ noktalarının yaklaşık koordinatları ile gözlemlerin ağırlıkları belirlendikten sonra dengeleme işlemine geçilebilir. Matris gösterimiyle dengeleme hesabı;

v = Ax-1	*Fonksiyonel Model*
I	*Stokastik Model*
$^{\wedge}11^{n}$	*Normal Denklemler*

Nx - n = 0 N

= A^T PA n =

A^T Pl

O = N^1

	Normal Denklemler Ters Matrisi
x = Qn	*Dengeleme Bilinmeyenleri*

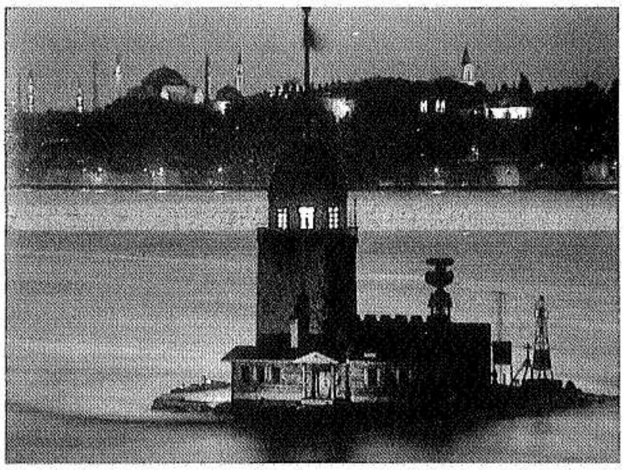
şeklinde yapılmaktadır. 14

Chamber of
Surveying
Engineers

Call for Abstracts

International
Federation of
Surveyors

ISTANBUL - 97



International Symposium on GIS/GPS

September 15-19, 1997

Harita ve Kadastro Mühendisleri Odası Dergisi 81. Sayı ekidir.

Bellek Gereksinimi:

m tane ölçünün *n* tane bilinmeyenin olduğu dengeleme problemini bilgisayarda yapabilmek için aşağıda boyutları verilen dizinli değişkenlere bellekte yer ayrılması gerekir;

vektörler: $l(m)$, $P(m)$, $V(m)$, $A^T Pl(n)$, ${}^x(n)$

matrisler: $A(m,n)$ $N(n,n)$, $Q(n,n)$

Burada görüldüğü gibi vektörlere ayrılan toplam $(3m+2n)$ elemanlık alan matrislere ayrılan toplam $(mn+2n^2)$ 'nin yanında gözardı edilebilecek durumdadır. Dengeleme hesabında her zaman $(m>n)$ olmakla beraber $(m=n)$ alırsak toplam bellek gereksinimi $(3n^2)$ eleman olacak söz konusu matrislerin *tek duyarlıklı (single precision)* olarak bilgisayar belleğinde saklanması uygundur. Bir reel sayı *4 byte'hk* alana yerleşeceğinden toplam bellek gereksinimi: $(J2n^2)$ *byte* olacaktır. Matrisleri *çift duyarlıklı (double precision)* tanımlamak bellek gereksinimini iki katına çıkaracağı ve işlem süresini de artıracağı için uygun değildir.

Bellek Gereksinimini Azaltmak İçin Neler Yapılabilir

1- Yöneltme Bilinmeyeninin İndirgenmesi:

Yatay doğrultuların gözlendiği bütün jeodezik ağlarda yatay doğrultu gözleminin yapıldığı her noktaya bir de yöneltme bilinmeyeni düşmektedir. Aynı noktada bir kaç kez gözlem yapılmışsa her defasında ayrı bir yöneltme bilinmeyeni kullanılır. Yöneltme bilinmeyeninin normal denklemler kurulmadan önce indirgenmesi bilinmeyen sayısını (matrisin boyutunu) dolayısı ile hesaplama süresini de azaltacaktır. Yöneltme bilinmeyeninin indirgenmesi için yaygın olarak *toplam denklem yöntemi* ve *schreiber yöntemleri* kullanılmaktadır. Toplam denklem yöntemi bilgisayarla çalışmaya daha uygundur. Schreiber yöntemi; A-hata denklemleri katsayılar matrisinin satır sayısını durak nokta sayısı kadar büyütmektedir.

2- *Zo* Yaklaşık Yöneltme Bilinmeyeninin Hesabı:

Zo'ın hesabı literatürde bir durak noktasında yapılan bütün yatay doğrultu gözlemlerinden yararlanarak

$$Zo = \frac{[t_o - r]}{n}$$

şeklinde ortalama olarak gerçekleştirilmektedir. Oysa *Zo* adı üzerinde bir yaklaşık değer olduğu için her durak noktasında yalnız ilk doğrultudan $Zo = to - r$ şeklinde hesaplanabilir. Eğer bu şekilde hesaplama yapılırsa ilk doğrultu için hesaplanacak -/ sabit terimi de otomatikman sıfır olacaktır. Bu şekilde *Zo* hesabının hiçbir sakıncası olmayıp yazılımı sadeleştirdiği gibi hesaplama süresini de azaltacaktır.

3- A- Hata Denklemleri Katsayılar Matrisini Kurmadan Direkt Normal Denklemler Katsayılar Matrisinin Kurulması:

Deneyimsiz programcılar A- matrisini kurduktan sonra *N = AJPA* matrisini kurmak için önce A^T transpoz matris, *P* ağırlık matrisi, ve bunların parçımı için A^TP matrisine bellekte yer ayırarak bellek gereksinimini gereksiz yere kat kat artırmakta, diğer bir deyişle maximum dengeleyebileceği ağ büyüklüğünü küçültmektedir. Ölçülerin korelasyonsuz (genellikle) olduğunda *P (m,m)* matrisi köşegen yapıda oluşacağı için bunun yerine *m* elemanlı bir vektör yeterli olacaktır. Af = A^TPA normal denklemler matrisi *A* matrisini oluşturmadan hesaplanabilir. (Bektaş, 1986) Bunun için her ölçüye ait düzeltme denklemi oluştururken (a^, bj_k katsayıları) bu denklemin normal denklemlere katkısı hesaplanır ve normal denklem katsayılarına eklenir. Görüldüğü gibi normal denklemler düzeltme denklemleri oluşturulurken kurulmaktadırlar. Normal denklemlerin simetrik oluşu nedeniyle yalnızca ana köşegen ve üstündeki terimler hesaplatılarak tek boyutlu bir dizide (vektörde) saklanır.

4- Normal Denklem Katsayılar Matrisinin Tersinin Hesabı:

Ters matris (invers) hesabı, dengeleme hesabında zamanın en fazla harcandığı kısımdır. Ters matris hesaplayan algoritmaların bazıları invers işlemini ikinci bir matris üzerinde yapmaktadır. Aynı matris üzerinde invers alma işlemini gerçekleştiren algoritmalar da vardır. Şüphesiz en uygun yöntem simetrik matrislerde yalnız simetrik ve köşegen üstündeki elemanların depolandığı tek boyutlu dizilerde (vektörde) invers alabilen algoritmadır. *Cholesky* algoritması buna olanak sağlar. *Cholesky.*yöntemiyle inversi alınacak matrisin *pozitif tanımlı* olması gerekir. Normal denklem katsayılar matrisi *N* her zaman pozitif tanımlı olduğu için N^l inversi cholesky yöntemiyle üç aşamada gerçekleştirilir.

1. adımda Af matrisi $N = C^TC$ şeklinde üst üçgen matris çarpanlarına ayrılır.

2. adımda *C* üst matrisinin C~^ inversi alınır.

3. adımda C^J üst üçgen matrisi kendi transpozuyla çarpılarak $N\sim^J = (C'^J)\ (C''^l)$^ şek linde N'^J ters matrisi elde edilir.

Cholesky yöntemi karekök alarak indirgeme yaptığı için yuvarlatma hatalarından da en az etkilenmektedir.

Tüm bu işlemler yapıldığında;

a-) Yöneltme bilinmeyenleri normal denklemler kurulmadan önce indirgeniyor.

b-) *A* matrisi kurulmadan doğrudan normal denklemlerin yalnız simetrik kısmı bir vektörde depolanıyor.

c-) Ters matris hesabı da aynı vektör üzerinde gerçekleştiriliyor.

Toplam bellek gereksinimi:

n: dengeleme bilinmeyen sayısı (yöneltme bilinmeyenleri hariç)

$4n\ (n+l)/2$ *byte* $= 2n(n+l)=2n^2$ *byte* olacaktır.

p tane koordinatları dengelenecek noktası olan bir nirengi ağı için *(n- 2p)* toplam bellek gereksinimi $8\ p^2$ *byte* olacaktır. Oysa klasik yöntemde bellek gereksinimi $48\ p^2$ *byte* olmaktadır. Görüldüğü gibi bellek gereksinimi hiç de azımsanmayacak şekilde altıda birine inmektedir.

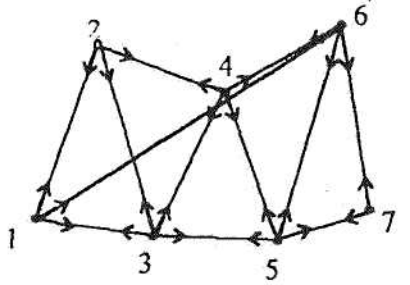
Aşağıdaki tabloda değişik nokta sayılarına göre bellek gereksiniminin klasik yöntem ile önerilen yöntemlerde ne olacağı görülmektedir.

Nokta Sayısı (p)	Klasik (byte)	Yöntem (Kb)	Önerilen (byte)	Yöntem (Kb)
10	4800	4.7	800	0.8
20	19200	18.8	3200	3.1
40	76800	75	12800	12.5
50	120000	117.2	20000	19.5
100	480000	468.8	80000	78.1

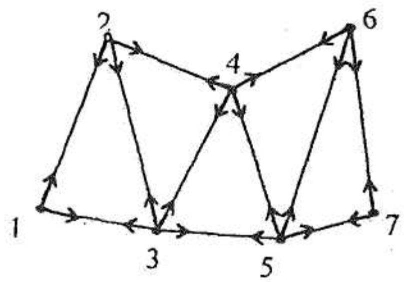
(1Kb=1024 byte)

Normal Denklemlerin Bantlaştırılmasi

Bu aşamadan sonra bellek gereksinimini azaltıcı daha neler yapılabilir diye düşünüldüğünde yapılacak tek şey normal denklemlerin bant yapılı hale getirilmesidir. Bant yapıdan kastedilen bilinmeyenlerin sırasını uygun şekilde düzenleyerek dolu elemanları (sıfır olmayanları) olabildiğince ana köşegene yakın paralellere getirmek sıfır elemanları oluşan bandın dışına çıkarılmasıdır. Ancak bantlaştırma işlemini yapmak her zaman mümkün değildir. Bantlaştırma işlemi; ağın şekli, gözlem planıyla doğrudan ilgilidir. Ağın bir ucundaki nokta ile diğer ucundaki nokta arasında gözlem bağlantısı varsa bantlaştırma işlemi gerçekleşemez. Bantlaştırma işlemi için en uygun ağ şekli birbirini kesmeyen üçgenlerden oluşan yüzey ağlarıdır.



şekil- (a)



şekil- (b)

Yukarıdaki nirengi ağlarından (a) daki ağ yapısı bantlaştırmaya olanak sağlamaktadır. Burada oluşacak bant matrisinin genişliği, maximum gözlemin yapıldığı noktadaki gözlem sayısımn (bakılan nokta sayısının) bir fazlasıdır. Örneğimizde 3,4 ve 5 nolu noktada 4'er gözlem yapıldığı için bant genişliği 4+1=5 nokta olacaktır. Nirengi ağında nokta başına iki koordinat bilinmeyeni düştüğü için bant genişliği 5x2=10 olacaktır, (b) deki ağ yapısı (a) ile hemen hemen aynı olmakla beraber 1-6 nolu noktalar arasında gözlem yapılması bantlaştırmaya olanak sağlamaz. 1-6 gözleminin ağdan çıkarılması da günümüz ölçme anlayışıyla bağdaşmaz. Yapılacak fazladan her gözlemin istatistik güveni artırdığı ve duyarlığıyla orantılı olarak sonuçları iyileştirdiği bir gerçektir.

Bantlaştırma işleminin büyük (ülke ve kıta) ağlarında uygulanması çok elverişli olmaktadır. Böylece normal denklemlerin bant genişliği dolu diğer elemanları sıfır olduğu için çok büyük bellek tasarrufu sağlanabilmektedir. Ancak bantlaştırma işleminin bir olumsuzluğu da bant matris üzerinde inverse olanak sağlamamasıdır. Bilindiği gibi matrislerin tersleri kendileriyle aynı karakterde ve aynı boyutta olmalarına rağmen (yani giriş-tersi alınacak matris simetrik, antisimetrik, ortogonal* köşegen veya üçgen yapıda ise matrisin tersi de aynı karakterde olmaktadır.) bant matrislerin tersi bant yapıda olmamaktadır. Bu nedenle giriş matrisinin yüklendiği vektör invers alma işleminde yetersiz kalmakta daha çok alana gerek sinim duyulmaktadır. Buradan bantlaştırma işlemi yapıldığında dengeleme bilinmeyenlerinin ters matris hesaplamadan gerçekleştirilmesi gerektiği dolayısı ile ters matris elemanlarının kullanıldığı duyarlık yapılamayacağı (im, my gibi) ortaya çıkmaktadır.

	1	2	3	4	5	6	7
1	X	X	X	0	0	0	0
2	X	X	X	X	0	0	0
3	X	X	X	X	X	0	0
4	0	X	X	X	X	X	0
5	0	0	X	X	X	X	X
6	0	0	0	X	X	X	X
7	0	0	0	0	X	X	X

Yukarıdaki tablo şekil -(a) daki 7 noktalı nirengi ağımız için normal denklemlerin dolu ve boş kısımlarını göstermektedir. Matris 14x14 = 196 elemanlı olmaktadır. Yalnız simetrik elemanların kullanılması durumunda 14x(14+l)/2=105 eleman olacaktır. Bant matrisin kullanılması durumunda 14+13+12+11+10+9=69 elemanlık bellek yeterli olacaktır. Bant matris elemanları satır veya köşegen düzeninde vektöre yüklenebilir. Bant matrisin *i. satır j.* sütün elemanı vektörün *k.* elemanı olur. *i, i, ve k* indisleri arasındaki ilişki

satır düzeninde yerleştirme

k=(i-l)(2n-i)/2+j* (*iij*)

1	2	3
	4	5
		6

köşegen düzenide yerleştirme

*k=n *(j-l)-(j-i) *(j-i-l)/2+i* (*i$j*)

1	4	6
	2	5
		3

SERBEST AĞ DENGELEMESİ

Büyük Ölçekli Haritaların Yapım Yönetmeliğinde (BÖHYY) jeodezik ağların serbest dengelenmesi gerektiği belirtilmektedir. Bilindiği gibi serbest ağ dengelemesinde tek boyutlu nivelman ve gravite ağlarında (1), yalnız doğrultuların ölçüldüğü nirengi ağlarında (4), eğer kenar ölçüsü de varsa (3), üç boyutlu jeodezik ağlarda (7), kenar ölçüsü de varsa (6) tane belirsiz datum parametresi olmaktadır. Datum parametresi sayısı (d) kadar bilinmeyen değişmez (sabit) alınmazsa kurulacak normal denklemler tekil (singular) yapıda olacağı için $Q = N"^1$ Cayley tersi hesaplanamaz. Bunun yerine $x^Tx=min.$ çözümünü veren pseudo ters matris $Q=N^+$ hesaplanarak çözüme gidilir. Pseudo ters matris hesabı için çeşitli yöntemler vardır. Bunlardan en kısa yöntem (Bektaş, 1990) da verilmiştir. Bellek tasarrufuna olanak sağlayan en uygun yöntem G benzerlik dönüşüm katsayılar matrisinden yararlanarak alınan inverstir. G matrisinin nasıl kurulacağı (Öztürk-Şerbetçi, 1992) de açıkça anlatılmıştır.

$$Q = N+ = (N+GG"^1)"^1 -GG^T$$

Bu yöntemde her ne kadar ekstradan G matrisi ve GG^T çarpım matrisi için ayrıca bellek gerektiriyor gibi gözüküyorsa da yazılımda yapılacak küçük bir değişiklikle G ve GG^T matrisleri kurulmadan bu sorun halledilebilir.

Vektör üzerindeki normal denklemlerin her (N)y elemanına (GG^T)y terimi hesaplatılarak eklenir. Tek boyutlu nivelman ve gravite ağları için (GG^T)ij terimleri sabit olup nokta sayısının tersine (l/p)'ye eşittir. $(N+GG^T)$ toplamı oluşturulduktan sonra bu toplamın $(N+GG^1)"^1$ caylev inversi bulunduğu vektör üzerinde hesaplanabilir bu hesaplanan inversten daha önce GG^T eklenmesi için yapılan işlemler bu kez çıkarılması için tekrarlanır. Böylece N^+ pseudo inversi hesaplanmış olur.

İterasyon İşleminin Yapılması

Gerek yaklaşık değerlerin iyi seçilmemesi ve gerekse de kötü kondüsyonlu büyük boyutlu matrislerin inverslerinin alınmasında oluşacak yuvarlatma hataları nedeniyle gerçek değerlere hemen ulaşılamaz ve dengeleme hesabı sonuç denetimleri de tam tutmaz. Bu durumda yapılması gereken dengeleme hesabı sonrası bulunan dengeli koordinatların yaklaşık koordinat olarak alınarak dengeleme hesabının .yinelenmesidir. Bu yineleme (iterasyon) işlemine dengeleme bilinmeyenleri sıfır oluncaya kadar devam edilir. Yapılan her iterasyon sayısının hesaplama süresini katlayarak artıracağı açıktır. Oysa ağ noktalarının koordinatlarındaki küçük (dengeleme hesabı sonucunda gelen) değişimlerin *(A)* hata denklemleri katsayılar matrisine dolayısı ile normal denklemlere $N = A^TPA$ yansımadığı bilinmektedir. Bu nedenle itersayonlu dengeleme işleminin yapılmasında her iterasyonda sadece -/ile $n=A^TPl$ hesaplatılır ve ilk bulunan invers matrisle çarpılarak bilinmeyenler gerçeğe çok uygun ve çok hızlı bir şekilde belirlenir, îterasyon işlemi özellikle serbest ağ dengelemelerinde gereklidir. İterasyonun yakınsamaması ya da çok geç yakınsaması modelimizde bir yanılgı olduğunun göstergesidir.

Düzeltmelerin Hesabı:

A- matrisi kurulmadığı için düzeltme denklemleri v = *A x -1* eşitliğinden hesaplanamaz. Düzeltmeler sonuç denetim eşitliklerinden tersten gelerek hesaplanır. Örneğin yatay doğrultular için bir *Pi* noktasında gözlenen r *y* doğrultusu için; Vi/ düzeltmesi şöyle hesaplanır.