

GPS DEFORMASYON AĞLARININ θ 2-ÖLÇÜTÜ İLE ANALİZİ : BİR PROGRAM VE UYGULAMA ÖRNEĞİ

Temel BAYRAK*

ÖZET

Bu makalede GPS deformasyon ağlarının θ 2-ölçütü yöntemine göre statik analizi için Fortran programlama dilinde yazılmış bir program tanıtımı yapılmaktadır. Çalışma kapsamında yöntemin teorisi, programın kullanımı hakkında gerekli bilgiler, bir uygulama örneği ve program listesi verilmiştir.

ABSTRACT

ANALYSIS OF GPS DEFORMATION NETWORKS BY θ 2-CRITERATION : A PROGRAM LIST AND AN APPLICATION OF IT

This paper introduces a Fortran program using θ 2-criterion to analyze GPS deformation networks. Theory of the method, information about use of the program, list of the program and an application of it were given in the article.

GİRİŞ

Statik model, bir objede hareket oluşup oluşmadığını, o obje ve çevresini kapsayan deformasyon ağının çeşitli periyotlarda belirlenen nokta koordinat farklarını istatistik olarak eş-değerlik testi ile saptayan en temel yöntemdir. Statik modelin konusu, deformasyon irdelemesi yapılan objenin karakteristik noktalarına ait deformasyon vektörlerinin, zamandan ve etkiyen dış kuvvetlerden bağımsız olarak belirlemektir. Bu modelle deformasyonun belirlenmesi için öncelikle obje çeşitli periyotlarda ölçülmeli ve her periyottaki ölçüler ayrı ayrı denetlenmelidir. Ölçme periyotları arasındaki koordinat farkları hem araştırılan objedeki deformasyonu hem de gözlemlerdeki hataları yansıtır. Bu nedenle periyotlar arasındaki koordinat farkları, istatistik yöntemlerle test edilerek deformasyon irdelemesi yapılır. Objeye etki eden

* tbayrak@ktu.edu.tr Karadeniz Teknik Üniversitesi Jeodezi ve Fotog. Müh. Bölümü Trabzon

yükler ve geometrik reaksiyonlar arasındaki davranışların incelenmesi statik modelin konusu dışındadır. Bu modelde tüm sistemin bir kez ölçülmesi sırasında noktaların sabit kaldığı varsayılır. Bu tanımıyla statik model jeodezik olarak en çok uygulanan bir deformasyon analizi yöntemidir. Basit olarak iki koordinat değerinin karşılaştırılması yerine, ölçülerin tamamının değerlendirildiği istatistik testlere dayalı çeşitli statik deformasyon modelleri geliştirilmiştir. Bu çalışmada statik modellerden uygulamada en çok kullanılan θ^2 -Ölçütü yöntemiyle deformasyon belirlemesi yapılmıştır.

Makalede, GPS ağırları için yöntemin teorisi verilmiş ve bu teoriye göre yazılmış bir program yardımıyla bir uygulama örneği sunulmuştur. Ayrıca programın listesinin yanı sıra kullanımıyla ilgili bilgiler de verilmiştir.

θ^2 -ÖLÇÜTÜ YÖNTEMİ

Deformasyon ağlarının ölçme planı ve uygulanan ölçme yöntemleri, araştırma süresi boyunca aynı kalmasına rağmen yöntemden kaynaklanan sistematik etkiler, karşılaştırılan periyotların tümünde benzer ve aynı büyüklükte ortaya çıkar. Eşdeğerlik testleri sırasında koordinat vektörlerinin farkları ile işlem yapıldığından belirlenen deformasyonları sistematik hatalardan arındırmak mümkündür. İki ölçme periyodu arasında geçen sürede hiçbir noktada deformasyon olmasa bile kaçınılmaz ölçü hataları ya da deformasyonlar nedeniyle, bu ölçülerden elde edilen nokta kümeleri arasında kesin bir eşdeğerlik beklenemez. İki ölçü periyodunda elde edilen nokta kümelerinin birbiri üzerine dönüştürülmesi sonucunda bazı noktalar az çok farklılıklar gösterecektir. Bu farklılığın rasgele hatalardan mı kaynaklandığı, yoksa karşılaştırılan bu iki konum bilgileri kümesi arasında bir değişim mi bulunduğu sorusuna θ^2 -Ölçütü ile çözüm aranmaktadır. θ^2 -Ölçütü yönteminde önce tüm periyotlardaki ölçüler ayrı ayrı serbest dengelenerek, dengeli koordinatlar vektörü $\underline{x}$, düzeltmelerin kareleri toplamı $\underline{v}^T \underline{p}$, bilinmeyenlerin ters ağırlıklı matrisi $\underline{Q}_{xx}$ hesaplanır. İki periyot arasında anlamlı nokta hareketi olup olmadığını belirlemek için, periyotlardan elde edilen dengeli koordinatların farkları ($\underline{d}$),

$$\underline{d} = \underline{x}_2 - \underline{x}_1 \quad (1)$$

olarak hesaplanır. Ağın herhangi bir noktasında deformasyon olup oluşmadığını araştırmak için, sıfır hipotezi hiçbir noktada deformasyon yoktur şeklinde

$$H_0 : \underline{x}_2 - \underline{x}_1 = 0 \quad (2)$$

(2)'ye göre kurulur. $\underline{d}$ fark vektörüne ait kofaktörler matrisi $\underline{Q}_d$, iki grup ölçülerin birbirinden bağımsız olduğu varsayımı ile hata yayılma kuralı uygulandığında,

$$\underline{Q}_d = \underline{Q}_{xx_1} + \underline{Q}_{xx_2} = (\underline{A}_1^T \underline{P}_1 \underline{A}_1)^* + (\underline{A}_2^T \underline{P}_2 \underline{A}_2)^* \quad (3)$$

olarak elde edilir. Hipotezin düzeltmelerin ağırlıklı kareleri toplamına etkisi olan θ^2 -ölçütü,

$$\theta^2 = (\underline{x}_2 - \underline{x}_1)^T \{ (\underline{A}_1^T \underline{P}_1 \underline{A}_1)^+ + (\underline{A}_2^T \underline{P}_2 \underline{A}_2)^+ \}^+ (\underline{x}_2 - \underline{x}_1) \quad (4)$$

biçiminde elde edilir. Q_d 'nin rangı, ağın geometrik şeklinin ve datum parametrelerinin her iki ölçme periyodunda da aynı kaldığı durumlarda

$$h = \text{rang} (\underline{Q}_{xx_1}) = \text{rang} (\underline{Q}_{xx_2}) = u_k - d_{\text{datum}} \quad (5)$$

biçiminde hesaplanır. Burada, u_k , ağdaki koordinat bilinmeyen sayısı, d_{datum} , datum parametreleri sayısıdır. Ölçülerin, aynı aletler, aynı meteorolojik koşullarda, aynı ölçme ekibince yapıldıkları varsayılırsa her iki ölçü dizisinin kuramsal varyansları eşit olur. Her iki ölçü kümesinin ortak standart sapması

$$s_0^2 = \sqrt{\frac{\underline{v}_1^T \underline{P}_1 \underline{v}_1 + \underline{v}_2^T \underline{P}_2 \underline{v}_2}{f_1 + f_2}} \quad (6)$$

$$f_i = n_i - u_i + d_{\text{datum}} ; \quad i = 1, 2$$

eşitliğinden hesaplanır. Eşdeğerlik testinin test büyüklüğü,

$$T = \frac{\theta^2}{s_0^2 h} \quad (7)$$

olarak hesaplanır. Test değeri T , F -Tablo değeri ile karşılaştırılır. $f = f_1 + f_2$ olmak üzere, $T < F_{h, f, 1-\alpha}$ ise H_0 hipotezi geçerlidir. Yani iki periyot arasında geçen sürede $s = 1 - \alpha$ kadar bir istatistik güvenle ağ noktalarında deformasyon gözlenmemiştir denilebilir. $T > F_{h, f, 1-\alpha}$ ise H_0 hipotezi geçerli değildir. Yani iki periyot arasında geçen sürede $s = 1 - \alpha$ kadar bir istatistik güvenle ağ noktalarından birinde ya da birkaçında deformasyon olduğuna karar verilir. Deformasyon olduğuna karar verildikten sonra, ağda hareket eden noktalar tek tek belirlenir. Bunun için ağ noktaları içinde her seferinde bir nokta hareketli, diğerleri sabit olduğu varsayımı ile nokta sayısı kadar θ^2 -ölçütü değeri hesaplanır. d fark vektörü ve Q_d kofaktörler matrisi aşağıdaki gibi alt matrislere ayrılır.

$$\underline{d} = \begin{vmatrix} \underline{d}_F \\ \underline{d}_B \end{vmatrix} \quad (8)$$

$$\underline{Q}_d = \begin{vmatrix} \underline{Q}_{FF} & \underline{Q}_{FB} \\ \underline{Q}_{BF} & \underline{Q}_{BB} \end{vmatrix} \quad (9)$$

Burada B , hareketli noktaları ve F , sabit noktaları göstermektedir. Q_d matrisinin tersi hesaplanarak ağırlık matrisi aşağıdaki biçimde

$$\underline{Q}_d^{-1} = \underline{P}_d = \begin{vmatrix} \underline{P}_{FF} & \underline{P}_{FB} \\ \underline{P}_{BF} & \underline{P}_{BB} \end{vmatrix} \quad (10)$$

elde edilmiş olur. Her nokta deformasyon kuşkusunu bulunan B noktası olarak ele alınır. Böylelikle her yeni adımda başka bir noktanın koordinatları, alt vektör $\underline{d}_B$ olarak yazılır. Bu durumda ağırlık nokta sayısı p olmak üzere, p kadar aykırılık etkisi hesaplanır. Alt matrisler Gauss yöntemi ile indirgenerek,

$$\bar{\underline{d}}_B = \underline{d}_B - \underline{P}_{BB}^{-1} \underline{P}_{BF} \underline{d}_F \quad (11)$$

$$\bar{\underline{P}}_{FF} = \underline{P}_{FF} - \underline{P}_{FB} \underline{P}_{BB}^{-1} \underline{P}_{BF} \quad (12)$$

değerleri hesaplanır. Bu değerler yardımıyla θ^2 değeri

$$\theta^2 = \underline{d}^T \underline{Q}_d^{-1} \underline{d} = \underline{d}_F^T \bar{\underline{P}}_{FF} \underline{d}_F + \bar{\underline{d}}_B^T \underline{P}_{BB} \bar{\underline{d}}_B \quad (13)$$

olarak bağımsız iki bileşene ayrılır. $\underline{d}_F^T \bar{\underline{P}}_{FF} \underline{d}_F$ sabit olduğu varsayılan noktaya ait aykırılık, $\bar{\underline{d}}_B^T \underline{P}_{BB} \bar{\underline{d}}_B$ hareketli olduğu varsayılan her noktaya ait aykırılıktır.

$$(\theta^2)_i = (\bar{\underline{d}}_B^T \underline{P}_{BB} \bar{\underline{d}}_B)_i \quad i = 1, 2, \dots, p \quad (14)$$

Burada p, nokta sayısıdır. Toplam aykırılıktaki payı en büyük olan $(\theta^2)_{\max} = \text{Max}((\theta^2)_i)$ olan noktada, $s = 1 - \alpha$ kadar istatistik güvenle deformasyon olduğuna karar verilir. Ağda deformasyon bulunana başka nokta bulunup bulunmadığını araştırmak için $\underline{d}$ ve $\underline{Q}_d$ 'ye bir S-dönüşümü yapılarak geriye kalan (p-1) noktadan yararlanarak yeni bir datum verilir. $\underline{d}$ ve $\underline{Q}_d$ 'de deformasyon noktasına ait elemanlar, en son satır ve sütunlara atılır. GPS ağları için ortogonal özellikli dönüşüm parametre katsayılar matrisi (G), p sayıda nokta için aşağıdaki biçimde oluşturulur.

$$\underline{G}^T = \begin{bmatrix} \frac{1}{\sqrt{p}} & 0 & 0 & \frac{1}{\sqrt{p}} & 0 & 0 & \dots & \frac{1}{\sqrt{p}} & 0 & 0 \\ 0 & \frac{1}{\sqrt{p}} & 0 & 0 & \frac{1}{\sqrt{p}} & 0 & \dots & 0 & \frac{1}{\sqrt{p}} & 0 \\ 0 & 0 & \frac{1}{\sqrt{p}} & 0 & 0 & \frac{1}{\sqrt{p}} & \dots & 0 & 0 & \frac{1}{\sqrt{p}} \end{bmatrix} \quad (15)$$

G matrisinden yararlanarak S dönüşüm matrisi aşağıdaki biçimde elde edilir

$$\underline{S}_i = \underline{I} - \underline{G} \underline{G}^T \quad (16)$$

Dönüşüm matrisi yardımıyla $\underline{d}$ ve $\underline{Q}_d$ matrisi

$$\underline{d}_i = \underline{S}_i \underline{d} \quad (17)$$

$$\underline{Q}_{di} = \underline{S}_i \underline{Q}_d \underline{S}_i^T \quad (18)$$

biçiminde dönüştürülür. i.dönüşümden sonra fark vektörü ve bunun ters ağırlık matrisi

biçiminde dönüştürülür. i.dönüşümünden sonra fark vektörü ve bunun ters ağırlık matrisi

$$\underline{d}_i = \begin{bmatrix} \underline{d}_D \\ \underline{d}_N \end{bmatrix} = \underline{S}_i \begin{bmatrix} \underline{d}_F \\ \underline{d}_B \end{bmatrix} = \underline{S}_i \underline{d} \quad (19)$$

$$\underline{Q}_{di} = \begin{bmatrix} \underline{Q}_{DD} & \underline{Q}_{DN} \\ \underline{Q}_{ND} & \underline{Q}_{NN} \end{bmatrix} \quad (20)$$

biçiminde alt matrislere ayrılır. D, datum noktalarını; N, datum dönüşümüne katılmayan noktaları göstermektedir. $\underline{d}_N$, toplam aykırılıktaki payları en büyük olan θ^2_{max} noktalarına ait $\underline{d}_B$ değerinin dönüşümünden sonraki değeridir. i. belirleme adımından sonra kalan aykırılık,

$$\theta^2_{kalan} = \underline{d}_D^T \underline{Q}_{DD}^+ \underline{d}_D \quad (21)$$

olarak hesaplanır. Serbestlik dereceleri, m, deformasyon olduğu belirlenen noktaya ilişkin koordinatların sayısı olmak üzere $h_D = h - m$, $f_D = f - 3$ biçiminde hesaplanır. Test büyüklüğü,

$$T_D = \frac{\theta^2_{kalan}}{s_0^2 \cdot h_D} > F_{h_D, f_D, 1-\alpha} \quad (22)$$

ise ağda hareketli nokta vardır. Yeni hareketli nokta, yukarıda açıklanan işlemler tekrar edilerek belirlenir. Bu işlemlere test büyüklüğü, tablo değerinde küçük oluncaya kadar devam edilir. Son bir S dönüşümü ile ağda sabit kalan noktalar P_D , deformasyon olduğu kanıtlanan noktalar P_N ve deformasyon büyüklükleri d_N elde edilir (Yalçınkaya, Tanır, 2000; Koch, 1999; Wolf, 1997; Öztürk, 1992, 1978; Gründig vd., 1985; Pelzer, 1985; Ayan, 1982; Dupraz vd., 1979; Mierlo, 1978).

PROGRAMIN TANITIMI ve BİR UYGULAMA

GPS ağlarında yapılan ölçülerden θ^2 -ölçütü yöntemiyle deformasyonların belirlendiği bilgisayar programının listesi Ek A'da verilmiştir. Program Windows altında çalışan ve maksimum belleği kullanabilen Fortran Visual Workbench v. 1.00 derleyicisinde yazılmıştır. Program, Şekil 1 ve Şekil 2'de örnek bir uygulama için verilmiş olan iki giriş dosyasındaki verileri kullanarak iki çıkış dosyası üretir. Giriş dosyaları, birinci ve ikinci ölçü periyotlarında yapılan dengelemelerden elde edilmiş varyans-kovaryans matrislerini, dengeli koordinatları, düzeltmelerin kareleri toplamlarını (v^Tpv) ve serbestlik derecelerini (fazla ölçü sayısı) içermektedir. Programın ürettiği çıkış dosyalarının birisi statik analiz sonuçlarına ait özet diğeri ise ayrıntılı çıkış bilgilerini sunmaktadır. Şekil 3'teki çıkış dosyası, Şekil 1 ve Şekil 2'de verilmiş periyodik ölçülerinin programla gerçekleştirilmiş statik analizine ait özet bilgi çıkış dosyasıdır. Program bu dosyaya bazı ara değerlerle birlikte deforme olan noktaları ve noktaların periyodik ölçüler sonucunda elde edilmiş deformasyon büyüklüklerini yazdırmaktadır. Ayrıntılı çıkış dosyasında tüm ara işlemlere ait yazdırmalar yapılmaktadır. Programı çalıştırmadan önce ağdaki nokta sayısına karşılık gelen 8. satırdaki NSO ve 39. satırdaki NS yeni uygulamalar için değiştirilmelidir. Kullanıcı giriş ve çıkış dosyalarının isimleri de değiştirebilir.

SONUÇLAR

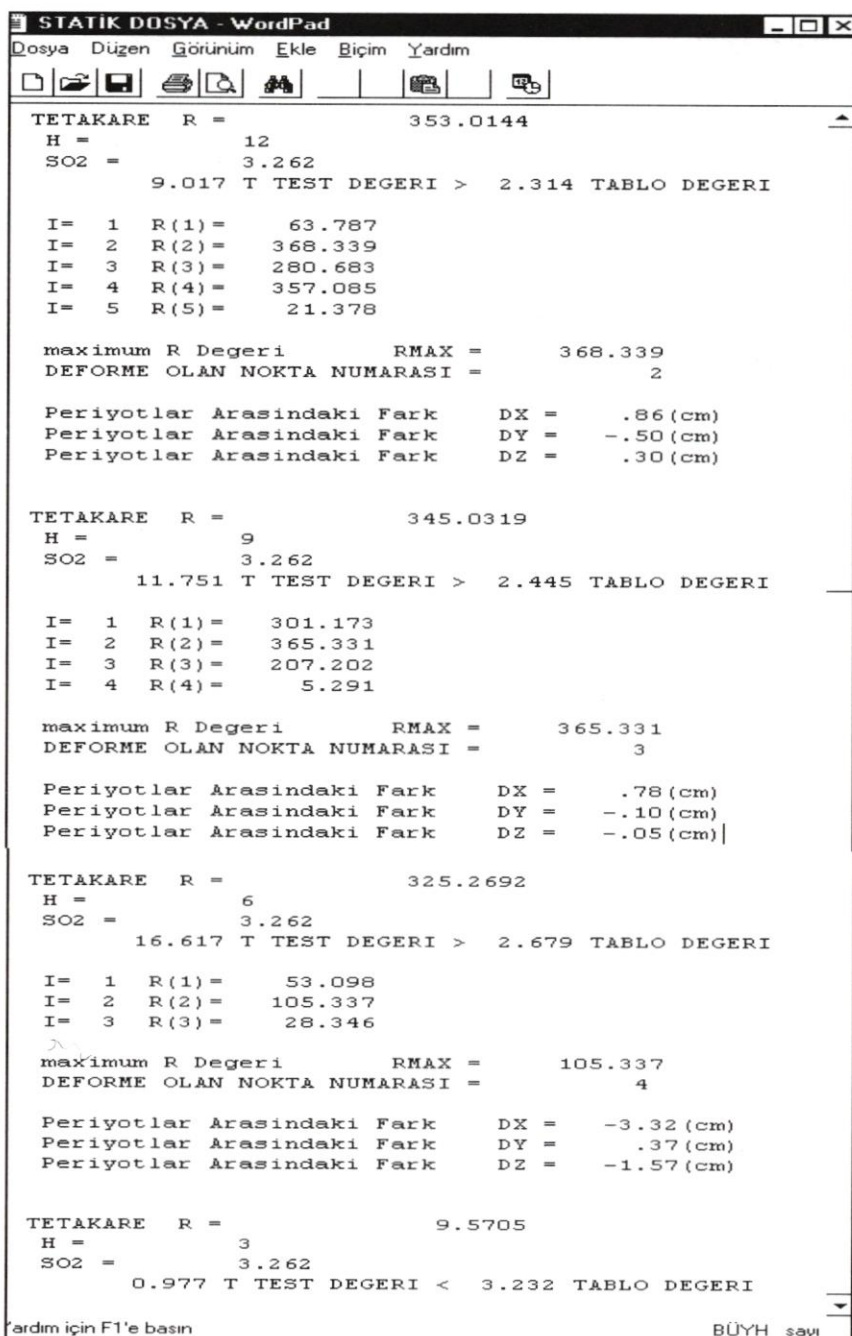
Program, deformasyonların belirlenmesi amacıyla tesis edilmiş olan jeodezik ağlarda periyodik olarak yapılmış GPS ölçülerinin statik analizine yönelik yapılmıştır. Yazılımda, statik modellerden ayırma gücü yüksek olan yani diğer bir deyişle çok küçük hareketleri saptayabilme algoritmasını içeren θ^2 -ölçütü yöntemi kullanılmıştır. Program, mühendislik uygulamaları için kullanışlı bir dil olan FORTRAN programlama diliyle yazılmıştır ve kullanımı kolaydır.

1.periyot - Not Defteri																			
Dosya	Düzen	Ara	Yardım																
.026	.020	.014	.000	-.003	.003	.003	.002	.004	-.015	-.011	-.010	-.014	-.008	-.011					
.020	.030	.010	-.004	-.006	.002	.003	.005	.003	-.012	-.018	-.007	-.007	-.011	-.009					
.014	.010	.024	.004	.002	.006	.003	.002	.007	-.009	-.006	-.021	-.012	-.009	-.017					
.000	-.004	.004	.057	.057	.022	.007	.003	.007	-.029	-.026	-.015	-.035	-.030	-.018					
-.003	-.006	.002	.057	.086	.017	.002	.008	.003	-.027	-.044	-.009	-.028	-.044	-.015					
.003	.002	.006	.022	.017	.044	.008	.003	.015	-.013	-.008	-.033	-.020	-.015	-.033					
.003	.003	.003	.007	.002	.008	.034	.028	.017	-.017	-.013	-.011	-.028	-.021	-.017					
.002	.005	.002	.003	.008	.003	.028	.043	.011	-.014	-.026	-.004	-.019	-.030	-.013					
.004	.003	.007	.007	.003	.015	.017	.011	.034	-.010	-.004	-.027	-.018	-.014	-.030					
-.015	-.012	-.009	-.029	-.027	-.013	-.017	-.014	-.010	.067	.061	.035	-.005	-.006	-.001					
-.011	-.018	-.006	-.026	-.044	-.008	-.013	-.026	-.004	.061	.094	.026	-.010	-.005	-.007					
-.010	-.007	-.021	-.015	-.009	-.033	-.011	-.004	-.027	.035	.026	.084	.002	-.005	-.003					
-.014	-.007	-.012	-.035	-.028	-.020	-.028	-.019	-.018	-.005	-.010	.002	.003	.066	.049					
-.008	-.011	-.009	-.030	-.044	-.015	-.021	-.030	-.014	-.006	-.005	-.005	.066	.092	.045					
-.011	-.009	-.017	-.018	-.015	-.033	-.017	-.013	-.030	-.001	-.007	-.003	.049	.045	.004					
1	3710292.4174	3084153.5423	4157861.5574																
2	3710205.6265	3084212.3164	4157845.5006																
3	3710274.8590	3084218.3802	4157799.2937																
4	3710389.3933	3084138.4622	4157788.8596																
5	3710501.7097	3084005.1004	4157907.9599																
12.360																			
9																			

Şekil 1: 1. Periyot veri dosyası

2.periyot - Not Defteri																
Dosya	Düzen	Ara	Yardım													
.017	.009	.011	-.007	-.006	-.006	-.004	-.002	-.003	-.004	-.001	-.002	-.001	.001	.001	.001	
.009	.015	.010	-.006	-.005	-.005	-.002	.002	-.001	-.001	-.006	-.002	.000	-.006	-.001		
.011	.010	.021	-.006	-.005	-.013	-.003	-.000	-.008	-.002	-.003	-.002	.000	-.001	.002		
-.007	-.006	-.006	.046	.031	.030	.011	.003	.007	-.024	-.013	-.015	-.025	-.014	-.016		
-.006	-.005	-.005	.031	.049	.031	.002	.002	.004	-.013	-.021	-.014	-.014	-.025	-.016		
-.006	-.005	-.013	.030	.031	.057	.006	.003	.015	-.015	-.013	-.029	-.016	-.016	-.030		
-.004	-.002	-.003	.011	.002	.006	.032	.017	.020	-.018	-.007	-.011	-.021	-.010	-.012		
-.002	.002	-.000	.003	.002	.003	.017	.023	.016	-.007	-.010	-.007	-.010	-.017	-.011		
-.003	-.001	-.008	.007	.004	.015	.020	.016	.038	-.011	-.007	-.021	-.013	-.011	-.024		
-.004	-.001	-.002	-.024	-.013	-.015	-.018	-.007	-.011	.052	.030	.034	-.005	-.008	-.005		
-.001	-.006	-.003	-.013	-.021	-.013	-.007	-.010	-.007	.030	.052	.031	-.007	-.014	-.006		
-.002	-.002	-.002	-.015	-.014	-.029	-.011	-.007	-.021	.034	.031	.056	-.004	-.006	-.003		
-.001	.000	.000	-.025	-.014	-.016	-.021	-.010	-.013	-.005	-.007	-.004	.054	.032	.033		
.001	-.006	-.001	-.014	-.025	-.016	-.010	-.017	-.011	-.008	-.014	-.006	.032	.063	.035		
.001	-.001	.002	-.016	-.016	-.030	-.012	-.011	-.024	-.005	-.006	-.003	.033	.035	.056		
1	3710292.4285		3084153.5374			4157861.5674										
2	3710205.6351		3084212.3113			4157845.5030										
3	3710274.8647		3084218.3802			4157799.2968										
4	3710389.3554		3084138.4679			4157788.8418										
5	3710501.7223		3084005.1047			4157907.9622										
46.360																
9																

Şekil 2: 2. Periyot veri dosyası



Şekil 3: q2-Ölçütü program sonuç dosyası

KAYNAKLAR

- Ayan, T., Jeodezik Ağlarda Deformasyon Analizine Genel Bakış, İ.T.Ü. Dergisi, 40, 21-25, İstanbul, 1982.
- Dupraz, H., Niemeier, W., Pelzer, H., Analyse von Deformationsmessungen durch Klaffungszerlegung, Deformationsanalysen, Heft 4, Hannover, 1979.
- Gründig, L. V., Nevreither, M., Bahndorf, J., Deformationanalyse und S-Transformation, ZFV, Heft 4, 151-160, 1985.
- Koch, K.-R., Parameter Estimation and Hypothesis Testing in Linear Models, Springer-Verlag Berlin, 1999.
- Mierlo, J., A Testing Procedure for Analysing Geodetic Deformation Measurements, Proceedings, Second International Symposium on Deformation Measurements by Geodetic Methods, Proceedings, 321-353, Bonn, 1978.
- Öztürk, E., Şerbetçi, M., Dengeleme Hesabı, K.T.Ü. Basımevi, Cilt III, Trabzon, 1992.
- Öztürk, E., Jeodezik Deformasyon Ölçülerinin İrdelenmesi θ^2 -Ölçütü, Harita Dergisi, 85, 44-52, 1978.
- Pelzer, H., Statische, Kinematische und Dynamische Punktfelder, in: Geodatische Netze in Landes-und Ingenieurvermessung II, PELZER, H., Stuttgart, 1985.
- Tanır, E., 2000, Deformasyon Analizinde Statik Değerlendirme Yöntemleri, Yüksek Lisans Tezi, K.T.Ü., Fen Bilimleri Enstitüsü, Trabzon.
- Wolf, P. R., Charles D., Ghilani: Adjustment Computation : Statistics and Last Squares in Surveying and GIS, John Wiley and Sons, Inc., ISBN 0-471-16833-5, 1997.
- Yalçınkaya (Ünver), M., Tanır, E., Determination of Movements on Mining Areas by Static Deformation Models, 11th International Congress of the International Society for Mine Surveying", Proceedings, 331-344, Cracow Poland, 2000.

EK A : PROGRAM LİSTESİ

C Hazırlayan : Arş.Gor.Temel BAYRAK (Niğde Üniversitesi)

Yoneten : Y. Doc. Dr. Mualla YALCINKAYA (K.T.Ü)

```
INTEGER I,J,K,L,NAS,H
REAL*8 PVV1,PVV2,X,Y,Z
REAL*8 X1,Y1,Z1,X2,Y2,Z2,QBIR,QIKI,D1,D2
REAL*8 SO2,T,DB,DF,R1,DT,RMAX
REAL*8 YEDEKDFARK,DFARK
```

C !!! NSO Ağdaki Nokta Sayısıdır. Kendi uygulamanız için değiştirmeniz gerekir
.....!!!

```
PARAMETER (NSO=5,MNO=3*NSO,YB=3,MY=MNO-YB)
DIMENSION QBIR(MNO,MNO),X1(NSO),Y1(NSO),Z1(NSO)
DIMENSION X(NSO),Y(NSO),Z(NSO)
DIMENSION QIKI(MNO,MNO),X2(NSO),Y2(NSO),Z2(NSO)
INTEGER NN1(NSO),NN2(NSO)
DIMENSION D1(MNO),D2(MNO),DFARK(MNO),QDD(MNO,MNO)
DIMENSION PDD(MNO,MNO),CY(MNO,MNO),YV(MNO),YV1(MNO)
DIMENSION DB(YB),DF(MY),DB1(YB),DB2(YB),YEDEKD(MNO)
DIMENSION PBB(YB,YB),PFF(MY,MY),PBF(YB,MY)
DIMENSION PBBINV(YB,YB),TA1(YB,YB),TA2(YB),TA3(YB)
DIMENSION CKONTROL(YB,YB),R1(NSO),DT(MNO),PTT(MNO,MNO)
DIMENSION YEDEKDFARK(MNO),YEDEKQDD(MNO,MNO),YQDD(MNO,MNO)
DIMENSION Y1QDD(MNO,MNO),Y2QDD(MNO,MNO),AQDD(MY,MY)
DIMENSION XAMKK(NSO),YAMKK(NSO),ZAMKK(NSO)
DIMENSION XNORM(NSO),YNORM(NSO),ZNORM(NSO),G(MY,6)
DIMENSION BIRIMAT(MY,MY),GTG(6,6)
DIMENSION GTGINV(6,6),GTA1(6,6),GTA2(6),GTA3(6)
DIMENSION GKONT(6,6),GS(MY,6),GS1(MY,MY),S_DONUSUM(MY,MY)
DIMENSION ATAMA(MNO,MNO),AMA(MNO,MNO),TAM(MNO),TAM1(MNO)
```

c *****

c 1 ve 2 numaralı noktalar veri dosyaları, 3 numaralı dosya ayrıntılı çıkış ve 4 numaralı dosya özet çıkış dosyasıdır. İsimleri

c kullanıcı kendi isteğine göre değiştirebilir

c *****

```
open(unit=1,status='old',file='ps_k0.DAT')
open(unit=2,status='old',file='ps_m1.DAT')
open(unit=3,status='old',file='p_k0m1_a.DAT')
open(unit=4,status='old',file='p_k0m1_c.DAT')
```

C *****

```
WRITE(3,10)
```

```
10 format(//,15x,'NOKTA KUMELERİ İÇİN ESDEĞERLİK TESTİ',//)
```

C !!! NS Ağıdaki Nokta Sayısıdır. Kendi uygulamanız için değiştirmeniz gerekir
.....!!!

```
DATA NS/5/
NAS=3*NS
DO 20 I=1,NAS
DO 20 J=1,NAS
READ(1,21)QBIR(I,J)
20 CONTINUE
21 FORMAT(F15.10)
DO 30 I=1,NS
READ(1,31)NN1(I),X1(NN1(I)),Y1(NN1(I)),Z1(NN1(I))
WRITE(3,31)NN1(I),X1(NN1(I)),Y1(NN1(I)),Z1(NN1(I))
30 CONTINUE
31 FORMAT(I9,3X,F12.4,6X,F12.4,6X,F12.4)
READ(1,*)PVV1
WRITE(3,*)PVV1
READ(1,*)IF1
WRITE(3,*)IF1
DO 35 I=1,NS
READ(1,36)NN1(I),X(NN1(I)),Y(NN1(I)),Z(NN1(I))
WRITE(3,36)NN1(I),X(NN1(I)),Y(NN1(I)),Z(NN1(I))
35 CONTINUE
36 FORMAT(I9,3X,F12.4,6X,F12.4,6X,F12.4)
DO 40 I=1,NAS
DO 40 J=1,NAS
READ(2,21)QIKI(I,J)
40 CONTINUE
DO 50 I=1,NS
READ(2,31)NN2(I),X2(NN2(I)),Y2(NN2(I)),Z2(NN2(I))
WRITE(3,31)NN2(I),X2(NN2(I)),Y2(NN2(I)),Z2(NN2(I))
50 CONTINUE
READ(2,*)PVV2
WRITE(3,*)PVV2
READ(2,*)IF2
WRITE(3,*)IF2
DO 60 I=1,NAS
D1(3*I-3+1)=X1(NN1(I))
D1(3*I-2+1)=Y1(NN1(I))
D1(3*I-1+1)=Z1(NN1(I))
60 CONTINUE
DO 70 I=1,NAS
D2(3*I-3+1)=X2(NN2(I))
D2(3*I-2+1)=Y2(NN2(I))
```

```

D2(3*I-1+1)=Z2(NN2(I))
70 CONTINUE
DO 80 I=1,NAS
DFARK(I)=(D2(I)-D1(I))*100
WRITE(3,81)I,D1(I),D2(I),DFARK(I)
80 CONTINUE
81 FORMAT(I6,5X,F12.4,3X,F12.4,3X,F12.4)
WRITE(3,*)' D FARK VEKTORU *****'
WRITE(4,*)' D FARK VEKTORU *****'
DO 85 I=1,NAS
YEDEKDFARK(I)=DFARK(I)
YEDEKD(I)=DFARK(I)
WRITE(3,82)I,YEDEKD(I)
WRITE(4,82)I,YEDEKD(I)
85 CONTINUE
82 FORMAT(I6,3X,F10.4)
DO 90 I=1,NAS
DO 90 J=1,NAS
QDD(I,J)=QBIR(I,J)+QIKI(I,J)
90 CONTINUE
91 FORMAT(2I6,3F15.10)
DO 93 I=1,NAS
DO 93 J=1,NAS
PDD(I,J)=QDD(I,J)
YEDEKQDD(I,J)=QDD(I,J)
93 CONTINUE
94 FORMAT(F15.10)
WRITE(3,*)'//////// QDD MATRISI ///////////'
CALL YAZ(QDD,NAS,NAS,MNO,MNO)
WRITE(3,*)'***** PDD MATRISI *****'
IR=3
CALL INVER3(PDD,CY,NAS,IR,MNO,YV,YV1)
CALL YAZ(PDD,NAS,NAS,MNO,MNO)
9997 WRITE(*,*)
R=0
DO 100 I=1,NAS
S=0
DO 110 J=1,NAS
110 S=S+DFARK(J)*PDD(J,I)
100 R=R+S*DFARK(I)
WRITE(3,101)R
WRITE(4,101)R
101 FORMAT('TETAKARE R =',F20.4)

```

```

H=0
H=NAS-3
WRITE(3,*)'H = ',H
WRITE(4,*)'H = ',H
SO2=0.
SO2=(PVV1+PVV2)/(IF1+IF2)
WRITE(3,*)'SO2 = ',SO2
WRITE(4,*)'SO2 = ',SO2
T=0.
T=R/(SO2*H)
WRITE(3,*)'T TEST DEGERI= ',T
IPAY=H
IPAYDA=IF1+IF2
CALL STATIS(0.05,IPAY,IPAYDA,2,XND,TABLO1)
WRITE(3,*)'ALFA = 0.05 IPAY = ',IPAY,' IPAYDA = ',IPAYDA
WRITE(3,*)'F-DAGILIM TABLO DEGERI = ',TABLO1
IF(T.GE.TABLO1) THEN
WRITE(3,*)
WRITE(3,*)T,'t TEST DEGERI ','>',TABLO1,' TABLO DEGERI OLDUGUNDAN'
WRITE(4,*)T,'t TEST DEGERI ','>',TABLO1,' TABLO DEGERI '
WRITE(3,*)
WRITE(3,*)' ##### UYARI #####'
WRITE(3,*)' AG NOKTALARININ BIRINDE'
WRITE(3,*)' YADA BIR KACINDA DEFORMASYON VARDIR'
WRITE(3,*)'#####'
ELSE
WRITE(3,*)T,' TEST DEGERI','<',TABLO1,' TABLO DEGERI OLDUGUNDAN'
WRITE(4,*)T,' TEST DEGERI','<',TABLO1,' TABLO DEGERI '
WRITE(3,*)' ##### AGDA DEFORMASYON YOK #####'
GO TO 9996
END IF
LL=NAS-5
IBOYUT=NAS
IARTIS=3
IARTIM=NS
9999 WRITE(*,*)
WRITE(3,*)NN1(IARTIM),' NOLU NOKTAYA AIT DFARK VEKTORU *****'
DO 280 I=1,NAS
280 WRITE(3,281)I,DFARK(I)
281 FORMAT(I6,3X,F10.4)
WRITE(3,*)NN1(IARTIM),' NOLU NOKTAYA AIT DB ALT VEKTORU ***'
DO 120 I=1,3
DB(I)=DFARK(NAS-3+I)

```

```

WRITE(3,121)DB(I)
120 CONTINUE
WRITE(3,*)NN1(IARTIM),' NOLU NOKTAYA AIT DF ALT VEKTORU ***'
DO 130 I=1,NAS-3
DF(I)=DFARK(I)
WRITE(3,121)DF(I)
130 CONTINUE
121 FORMAT(F10.4)
WRITE(3,*)NN1(IARTIM),'NOLU NOKTAYA AIT PDD MATRISI *****'
CALL YAZ(PDD,NAS,NAS,MNO,MNO)
WRITE(3,*)NN1(IARTIM),' NOLU NOKTAYA AIT PBB ALT MATRISI ***'
DO 140 I=1,3
DO 140 J=1,3
PBB(I,J)=PDD(NAS-3+I,NAS-3+J)
PBBINV(I,J)=PBB(I,J)
140 CONTINUE
CALL YAZ(PBB,3,3,3,3)
DO 150 I=1,NAS-3
DO 150 J=1,NAS-3
PFF(I,J)=PDD(I,J)
150 CONTINUE
DO 155 I=1,3
DO 155 J=1,NAS-3
PBF(I,J)=PDD(NAS-3+I,J)
155 CONTINUE
WRITE(3,*)'***** PBBINV PBB MATRISININ INVERSI *****'
IRP=0
IP=3
CALL INVER3(PBBINV,TA1,IP,IRP,3,TA2,TA3)
CALL YAZ(PBBINV,3,3,3,3)
WRITE(3,*)'***** KONTROL PBBINV*PBB=BIRIM MATRIS *****'
DO 170 I=1,3
DO 170 J=1,3
CKONTROL(I,J)=0
DO 170 K=1,3
CKONTROL(I,J)=CKONTROL(I,J)+PBBINV(I,K)*PBB(K,J)
170 CONTINUE
CALL YAZ(CKONTROL,3,3,3,3)
WRITE(3,*)'***** INDIRGENMIS DB MATRISI *****'
DO 180 I=1,3
SI=0
DO 190 J=1,NAS-3
TI=0

```

```

DO 200 K=1,3
200 TI=TI+PBBINV(I,K)*PBF(K,J)
190 SI=SI+TI*DF(J)
180 DB1(I)=SI
DO 210 I=1,3
210 DB2(I)=DB(I)-DB1(I)
WRITE(3,*)'***** DB(I) DB1(I) DB2(I) indirgenmis DB ***'
DO 220 I=1,3
WRITE(3,221)I,DB(I),DB1(I),DB2(I)
220 CONTINUE
221 FORMAT(I6,3X,3F10.4)
WRITE(3,*)'***** R=DBT*PBB*DB *****'
RI=0
DO 230 I=1,3
SI=0
DO 240 J=1,3
SI=SI+DB2(J)*PBB(I,J)
240 RI=RI+SI*DB2(I)
230 CONTINUE
R1(IARTIM)=RI
WRITE(3,*)'IARTIM=',IARTIM,' R1(',IARTIM,')=',R1(IARTIM)
IF(LL.LT.0) GO TO 9998
WRITE(3,*)** DFARK VEKTORUNDE YER DEGISTIRMELER YAPILIYOR **'
WRITE(3,*)** DFARK VEKTORU YENIDEN OLUSTURULUYOR **'
DO 250 I=LL,LL+2+IARTIS-3
250 DT(I)=DFARK(I+3)
DO 260 I=IBOYUT-2,IBOYUT
260 DT(I)=DFARK(I-IARTIS)
DO 270 I=LL,IBOYUT
270 DFARK(I)=DT(I)
WRITE(3,*)'PDD MATRISINDE YER DEGISTIRMELER YAPILIYOR'
DO 275 I=1,IBOYUT
DO 275 J=1,IBOYUT
PTT(I,J)=0
275 CONTINUE
DO 290 I=LL,LL+2+IARTIS-3
DO 290 J=1,IBOYUT
290 PTT(I,J)=PDD(I+3,J)
DO 300 I=IBOYUT-2,IBOYUT
DO 300 J=1,IBOYUT
300 PTT(I,J)=PDD(I-IARTIS,J)
DO 333 I=LL,IBOYUT
DO 333 J=1,IBOYUT

```

```

333 PDD(I,J)=PTT(I,J)
DO 310 I=LL,IBOYUT
DO 310 J=LL,IBOYUT-3
310 PTT(I,J)=PDD(I,J+3)
DO 320 I=LL,IBOYUT
DO 320 J=IBOYUT-2,IBOYUT
320 PTT(I,J)=PDD(I,J-IARTIS)
DO 330 I=LL,IBOYUT
DO 330 J=1,IBOYUT
330 PDD(I,J)=PTT(I,J)
LL=LL-3
IARTIS=IARTIS+3
IARTIM=IARTIM-1
IF(IARTIM.GT.0) GO TO 9999
9998 WRITE(*,*)
WRITE(3,*)'***** R DEGERLERI *****'
WRITE(4,*)
DO 338 I=1,NS
WRITE(3,*)I='I',NN('I')='I',NN1(I),R('I')='R',R1(I)
WRITE(4,*)I='I',R('I')='R',R1(I)
338 CONTINUE
WRITE(3,*)RMAX DEGERININ BULUNMASI
%% %% %% %% %% %% %% %% %% %% %% %% %% %% %% %% %%
ISAKLA1=NN1(1)
RMAX=R1(1)
DO 340 I=2,NS
IF(RMAX.LT.R1(I)) THEN
ISAKLA1=NN1(I)
RMAX=R1(I)
END IF
340 CONTINUE
WRITE(3,*)
WRITE(3,*)'maximum R Degeri RMAX =' ,RMAX
WRITE(3,*)'DEFORME OLAN NOKTA NUMARASI =' ,ISAKLA1
WRITE(3,*)
WRITE(3,346)YEDEKDFARK(3*ISAKLA1-3+1)
WRITE(3,347)YEDEKDFARK(3*ISAKLA1-2+1)
WRITE(3,348)YEDEKDFARK(3*ISAKLA1-1+1)
WRITE(4,*)
WRITE(4,*)'maximum R Degeri RMAX =' ,RMAX
WRITE(4,*)'DEFORME OLAN NOKTA NUMARASI =' ,ISAKLA1
WRITE(4,*)
WRITE(4,346)YEDEKDFARK(3*ISAKLA1-3+1)

```

```

WRITE(4,347)YEDEKDFARK(3*ISAKLA1-2+1)
WRITE(4,348)YEDEKDFARK(3*ISAKLA1-1+1)
346 FORMAT(' Periyotlar Arasindaki Fark DX =',F10.2,'(cm)')
347 FORMAT(' Periyotlar Arasindaki Fark DY =',F10.2,'(cm)')
348 FORMAT(' Periyotlar Arasindaki Fark DZ =',F10.2,'(cm)')
WRITE(4,*)
WRITE(3,*)'***** YEDEKDFARK VEKTORU *****'
WRITE(4,*)'***** YEDEKDFARK VEKTORU *****'
DO 403 I=1,IBOYUT
WRITE(3,413)YEDEKDFARK(I)
403 CONTINUE
413 FORMAT(F10.4)
DO 420 I=1,NS-1
IF(I.GE.ISAKLA1) THEN
DFARK(3*I-3+1)=YEDEKDFARK(3*I-3+1+3)
DFARK(3*I-2+1)=YEDEKDFARK(3*I-2+1+3)
DFARK(3*I-1+1)=YEDEKDFARK(3*I-1+1+3)
ELSE
DFARK(3*I-3+1)=YEDEKDFARK(3*I-3+1)
DFARK(3*I-2+1)=YEDEKDFARK(3*I-2+1)
DFARK(3*I-1+1)=YEDEKDFARK(3*I-1+1)
END IF
420 CONTINUE
DO 425 I=1,NAS-3
YEDEKDFARK(I)=0
425 CONTINUE
WRITE(3,*)ISAKLA1,' NOLU NOKTASI ATILMIS DFARK VEKTORU *****'
WRITE(4,*)ISAKLA1,' NOLU NOKTASI ATILMIS DFARK VEKTORU *****'
DO 430 I=1,NAS-3
YEDEKDFARK(I)=DFARK(I)
WRITE(3,435)I,DFARK(I),YEDEKDFARK(I)
WRITE(4,435)I,DFARK(I)
430 CONTINUE
435 FORMAT(I6,3X,F10.5,3X,F10.5)
DO 500 I=1,NS-1
DO 510 J=1,NAS
IF(I.GE.ISAKLA1) THEN
YQDD(3*I-3+1,J)=YEDEKQDD(3*I-3+1+3,J)
YQDD(3*I-2+1,J)=YEDEKQDD(3*I-2+1+3,J)
YQDD(3*I-1+1,J)=YEDEKQDD(3*I-1+1+3,J)
ELSE
YQDD(3*I-3+1,J)=YEDEKQDD(3*I-3+1,J)
YQDD(3*I-2+1,J)=YEDEKQDD(3*I-2+1,J)

```

```

YQDD(3*I-1+1,J)=YEDEKQDD(3*I-1+1,J)
END IF
510 CONTINUE
500 CONTINUE
DO 515 I=1,NAS-3
DO 516 J=1,NAS
Y1QDD(I,J)=YQDD(I,J)
516 CONTINUE
515 CONTINUE
DO 520 I=1,NS-1
DO 530 J=1,NAS-3
IF(J.GE.(ISAKLA1*3-3+1)) THEN
Y2QDD(3*I-3+1,J)=Y1QDD(3*I-3+1,J+3)
Y2QDD(3*I-2+1,J)=Y1QDD(3*I-2+1,J+3)
Y2QDD(3*I-1+1,J)=Y1QDD(3*I-1+1,J+3)
ELSE
Y2QDD(3*I-3+1,J)=Y1QDD(3*I-3+1,J)
Y2QDD(3*I-2+1,J)=Y1QDD(3*I-2+1,J)
Y2QDD(3*I-1+1,J)=Y1QDD(3*I-1+1,J)
END IF
530 CONTINUE
520 CONTINUE
DO 540 I=1,NAS-3
DO 550 J=1,NAS-3
AQDD(I,J)=Y2QDD(I,J)
550 CONTINUE
540 CONTINUE
WRITE(3,*)'***** ILGILI SATIR VE SUTUN ATILMIS *****'
WRITE(3,*)'%%%%%%%%% QDD MATRISI %%%%%%%%%5'
CALL YAZ(AQDD,NAS-3,NAS-3,MY,MY)
DO 579 I=1,NS
WRITE(3,*)'ISAKLA1=',ISAKLA1
WRITE(3,*)'I=',I,'NN1(',I,')=',NN1(I),'NN2(',I,')=',NN2(I)
579 CONTINUE
DO 580 I=1,NS-1
IF(I.GE.ISAKLA1) THEN
X1(I)=X1(I+1)
Y1(I)=Y1(I+1)
Z1(I)=Z1(I+1)
X2(I)=X2(I+1)
Y2(I)=Y2(I+1)
Z2(I)=Z2(I+1)
ELSE

```

```

X1(I)=X1(I)
Y1(I)=Y1(I)
Z1(I)=Z1(I)
X2(I)=X2(I)
Y2(I)=Y2(I)
Z2(I)=Z2(I)
END IF
580 CONTINUE
WRITE(4,*)
DO 590 I=1,NS-1
WRITE(3,*)I=',I','NN1(',I,')=',NN1(I),'NN2(',I,')=',NN2(I)
WRITE(4,591)I,X1(I),Y1(I),Z1(I)
WRITE(4,591)I,X2(I),Y2(I),Z2(I)
590 CONTINUE
591 FORMAT(I4,3F15.4)
NS=NS-1
WRITE(3,*)'NOKTA SAYISI = ',NS
NAS=NAS-3
WRITE(3,*)'MATRIS BOYUTU = ',NAS
DO 750 I=1,NS
DO 740 J=1,3
IF(J.EQ.1) THEN
G((J+I*3-3),1)=1/sqrt(NS)
G((J+I*3-3),2)=0
G((J+I*3-3),3)=0
END IF
IF(J.EQ.2) THEN
G((J+I*3-3),1)=0
G((J+I*3-3),2)=1/sqrt(NS)
G((J+I*3-3),3)=0
END IF
IF(J.EQ.3) THEN
G((J+I*3-3),1)=0
G((J+I*3-3),2)=0
G((J+I*3-3),3)=1/sqrt(NS)
END IF
740 CONTINUE
750 CONTINUE
CALL YAZ(G,NAS,3,MY,3)
DO 760 I=1,NAS
DO 770 J=1,NAS
BIRIMAT(I,J)=0
IF(I.EQ.J) BIRIMAT(I,J)=1

```

```

770 CONTINUE
760 CONTINUE
CALL YAZ(BIRIMAT,NAS,NAS,MY,MY)
DO 870 I=1,NAS
DO 880 J=1,NAS
S=0
DO 890 K=1,3
S=S+G(I,K)*G(J,K)
890 CONTINUE
GS1(I,J)=S
880 CONTINUE
870 CONTINUE
WRITE(3,*)'S_DONUSUM MATRISI OLUSTURULUYOR **** *'
DO 900 I=1,NAS
DO 910 J=1,NAS
S_DONUSUM(I,J)=BIRIMAT(I,J)-GS1(I,J)
910 CONTINUE
900 CONTINUE
WRITE(3,*)'##### S_DONUSUM MATRISI #####'
CALL YAZ(S_DONUSUM,NAS,NAS,MY,MY)
WRITE(3,*)'*** DONUSTURULMUS KOORDINAT FARKLARI VEKTORU ****'
WRITE(4,*)'*** DONUSTURULMUS KOORDINAT FARKLARI VEKTORU ****'
DO 920 I=1,NAS
DFARK(I)=0
920 CONTINUE
DO 930 I=1,NAS
T=0
DO 940 J=1,NAS
T=T+S_DONUSUM(I,J)*YEDEKDFARK(J)
940 CONTINUE
DFARK(I)=T
930 CONTINUE
DO 950 I=1,NAS
YEDEKDFARK(I)=DFARK(I)
WRITE(3,955)I,DFARK(I),YEDEKDFARK(I)
WRITE(4,955)I,DFARK(I)
950 CONTINUE
955 FORMAT(I6,3X,F10.5,3X,F10.5)
WRITE(3,*)' DONUSTURULMUS VARYANS-KOVARYANS (QDD) MATRISI '
DO 960 I=1,NAS
DO 970 J=1,NAS
T=0.
DO 980 K=1,NAS

```

```

S=0.
DO 990 L=1,NAS
990 S=S+S_DONUSUM(I,L)*AQDD(L,K)
980 T=T+S*S_DONUSUM(J,K)
970 QDD(I,J)=T
960 CONTINUE
CALL YAZ(QDD,NAS,NAS,MNO,MNO)
DO 996 I=1,MY
DO 996 J=1,MY
AQDD(I,J)=0
S_DONUSUM(I,J)=0
996 CONTINUE
DO 1000 I=1,NAS
DO 1010 J=1,NAS
ATAMA(I,J)=QDD(I,J)
YEDEKQDD(I,J)=QDD(I,J)
1010 CONTINUE
1000 CONTINUE
WRITE(3,*)'***** PDD MATRISI *****'
IRK=3
CALL INVER3(ATAMA,AMA,NAS,IRK,MNO,TAM,TAM1)
DO 1200 I=1,NAS
DO 1200 J=1,NAS
PDD(I,J)=ATAMA(I,J)
1200 CONTINUE
CALL YAZ(PDD,NAS,NAS,MNO,MNO)
DO 1300 I=1,NAS
TAM(I)=0
TAM1(I)=0
DO 1300 J=1,NAS
ATAMA(I,J)=0
AMA(I,J)=0
1300 CONTINUE
GO TO 9997
9996 WRITE(*,*)
close(unit=1)
close(unit=2)
close(unit=3)
close(unit=4)
END
C *****
SUBROUTINE YAZ(D,IST,ISUT,IST1,IST2)
DIMENSION D(IST1,IST2)

```

```

ISUTUN=ISUT
ISON=6
IF(ISUTUN.LE.ISON) ISON=ISUTUN
IBAS=1
IDON=0
404 IDON=IDON+ISON
409 WRITE(3,159) (J,J=IBAS,IDON)
159 FORMAT(1X,'Sat_r',I6,5I10)
WRITE(3,459)
459 FORMAT(1X,'Sutun',1X,65('.'))
DO 405 I=1,IST
405 WRITE(3,458) I,(D(I,J),J=IBAS,IDON)
458 FORMAT(I6,1X,6(F10.5))
IBAS=IBAS+ISON
IF(IBAS.GT.ISUTUN) GO TO 406
ISONY=ISUTUN-IDON
IF(ISONY.GT.ISON) GO TO 404
ISON=ISUTUN-IDON
GO TO 404
406 RETURN
END

```

C *****

```

SUBROUTINE INVER3 (A,B,N,ID,M,C,D)
DIMENSION A(M,M),B(M,M),C(M),D(M)
1 IR=N-ID
DO 2 I=1,IR
C(I)=1.
DO 2 J=1,IR
2 B(I,J)=A(I,J)
CALL CHOLIN (B,C,D,M,IR,ITOP,DET)
IF(ITOP-1) 4,3,4
3 IF(ID.EQ.N) STOP
ID=ID+1
WRITE(*,*) ' RANG BOZUKLUGU',ID
GO TO 1
4 WRITE(*,*) ' TERS ALMA ISLEMI BASLADI'
DO 5 I=1,IR
DO 5 J=1,IR
5 A(I,J)=B(I,J)
IF(ID.EQ.0) RETURN
DO 7 I=1,IR
DO 7 J=1,ID
T=0.

```

```

DO 6 K=1,IR
6 T=T+A(I,K)*A(K,J+IR)
7 B(I,J)=T
DO 8 I=1,IR
DO 8 J=1,ID
8 A(I,J+IR)=B(I,J)
DO 11 I=1,ID
DO 10 J=1,ID
T=0.
DO 9 K=1,IR
9 T=T+A(K,I+IR)*A(K,J+IR)
10 B(I,J)=T
11 B(I,I)=B(I,I)+1.
CALL CHOLIN (B,C,D,M,ID,ITOP,DET)
IF(ITOP.EQ.1) STOP
DO 13 I=1,IR
DO 13 J=1,IR
T=0.
DO 12 K=1,ID
12 T=T+A(I,K+IR)*B(K,J)
13 B(I+ID,J)=T
DO 15 I=1,IR
DO 15 J=1,ID
T=0.
DO 14 K=1,IR
14 T=T+A(I,K)*B(K+ID,J)
15 A(J+IR,I)=T
DO 17 I=1,IR
DO 17 J=1,IR
T=0.
TT=0.
DO 16 K=1,ID
T=T+A(I,K+IR)*A(K+IR,J)
16 TT=TT+A(K+IR,I)*A(J,K+IR)
17 A(I,J)=A(I,J)-T*TT
DO 19 I=1,ID
DO 19 J=1,ID
T=0.
DO 18 K=1,IR
18 T=T+A(I+IR,K)*B(K+ID,J)
19 A(I+IR,J+IR)=T
DO 21 I=1,IR
DO 21 J=1,ID

```

```

T=0.
DO 20 K=1,ID
20 T=T+A(I,K+IR)*A(J+IR,K+IR)
21 B(I,J)=T
DO 23 I=1,IR
DO 23 J=1,IR
T=0.
DO 22 K=1,ID
22 T=T+B(I,K)*A(J,K+IR)
23 A(I,J)=A(I,J)+T
DO 24 I=1,IR
DO 24 J=1,ID
24 B(I,J)=A(I,J+IR)
DO 26 I=1,IR
DO 26 J=1,ID
T=0.
DO 25 K=1,IR
25 T=T+A(I,K)*B(K,J)
A(I,J+IR)=T
26 A(J+IR,I)=T
DO 28 I=1,ID
DO 28 J=1,ID
T=0.
DO 27 K=1,IR
27 T=T+B(K,I)*A(K,J+IR)
28 A(I+IR,J+IR)=T
RETURN
END
SUBROUTINE CHOLIN (A,B,R,M,N,IS,DET)
REAL *8 S
DIMENSION A(M,M),B(M),R(M)
DO 110 I=1,N
IM1=I-1
IP1=I+1
S=A(I,I)
IF(IM1) 25,25,10
10 DO 20 J=1,IM1
20 S=S-A(J,I)**2
25 IF(S-1.E-38) 280,280,30
30 A(I,I)= SQRT(S)
IF(IP1-N) 40,40,75
40 DO 70 K=IP1,N
S=A(I,K)

```

```

IF(IM1) 70,70,50
50 DO 60 J=1,IM1
60 S=S-A(J,I)*A(J,K)
70 A(I,K)=S/A(I,I)
75 S=B(I)
IF(IM1) 100,100,80
80 DO 90 J=1,IM1
90 S=S-A(J,I)*B(J)
100 B(I)=S/A(I,I)
110 CONTINUE
DO 150 I=1,N
L=N-I+1
LP1=L+1
S=B(L)
IF(LP1-N) 120,120,140
120 DO 130 J=LP1,N
130 S=S-A(L,J)*B(J)
140 B(L)=S/A(L,L)
150 CONTINUE
DET=0.
DO 155 I=1,N
R(I)=A(I,I)
155 DET=DET+ ALOG10(R(I))
DET=DET*2.
DO 230 K=1,N
DO 230 I=K,N
IM1=I-1
IF(I-K) 170,160,170
160 S=1.
GO TO 230
170 S=0.
DO 200 J=K,IM1
200 S=S-A(J,I)*A(J,K)
230 A(I,K)=S/R(I)
DO 260 K=1,N
NK=N+K
DO 260 I=K,N
L=NK-I
S=A(L,K)
LP1=L+1
IF(LP1-N) 240,240,260
240 DO 250 J=LP1,N
250 S=S-A(L,J)*A(J,K)

```

```

260 A(L,K)=S/R(L)
DO 270 I=1,N
DO 270 K=1,I
270 A(K,I)=A(I,K)
IS=0
RETURN
280 IS=1
WRITE (*,281)
281 FORMAT(/5X, '***** BU ASAMADA MATRIS SINGULERDIR ')
RETURN
END
SUBROUTINE STATIS (ALFA,IS,ISE,ITANIM,X,TABLO)
DIMENSION Z(9)
T=SQRT(ALOG(1./(ALFA*ALFA)))
X=T-(2.515517+0.802853*T+0.010328*T*T)/
. (1.+1.432788*T+0.189269*T*T+0.001308*T*T*T)
IF(ITANIM-2) 1,3,4
1 Z(1)=X
DO 2 I=3,9,2
2 Z(I)=Z(I-2)*X*X
G1=(Z(3)+Z(1))/4.
G2=(5.*Z(5)+16.*Z(3)+3.*Z(1))/96.
G3=(3.*Z(7)+19.*Z(5)+17.*Z(3)-15.*Z(1))/384.
G4=(79.*Z(9)+776.*Z(7)+1482.*Z(5)-1920.*Z(3)-945.*Z(1))/92160.
TABLO=X+G1/IS+G2/(IS*IS)+G3/(IS*IS*IS)+G4/(IS*IS*IS*IS)
RETURN
3 S=1./IS
E=1./ISE
G=SQRT(0.5*(S+E))
C=X*G-(0.5*(S-E))
TABLO=EXP(2.*C)
RETURN
4 T=2./(9.*IS)
H=1.-T+X*SQRT(T)
TABLO=IS*H**3
RETURN
END

```