NASA Contractor Report 3793

Dynamics of Planetary Gear Trains

R. August, R. Kasuba,
J. L. Frater, and A. Pintz

GRANT NAG3-186
JUNE 1984

NASA

NASA Contractor Report 3793

Dynamics of Planetary Gear Trains

R. August, R. Kasuba,
J. L. Frater, and A. Pintz
Cleveland State University
Cleveland, Ohio

Prepared for
Lewis Research Center
under Grant NAG3-186

National Aeronautics
and Space Administration

**Scientific and Technical
Information Branch**

1984

TABLE OF CONTENTS

CHAPTER I

INTRODUCTION

1.1 General Remarks

Epicyclic gearing arrangements are comprised of four different elements that produce a wide range of speed ratios in a compact layout. These elements are: (1) Sun gear, an externally toothed ring gear co-axial with the gear train; (2) Annulus, an internally toothed ring gear co-axial with the gear train; (3) Planets, externally toothed gears which mesh with the sun and annulus; and (4) Planet Carrier, a support structure for the planets, co-axial with the train. The name "epicyclic" is derived from the curve traced by a point on the circumference of a circle as it rolls on the circumference of a second fixed circle.

By fixing one of the co-axial members and using the remaining two for input and output, three types of simple single-stage epicyclic gearing are possible. Generally, these are called planetary, star, and solar arrangements. This investigation is primarily concerned with planetary gear drives (Fig. 1) which have a fixed annulus with the planet carrier rotating in the same direction as the sun gear.

The principal advantages of epicyclic gears over parallel shaft gears are considerable savings in weight and space (see Table 1). These advantages stem from the fact

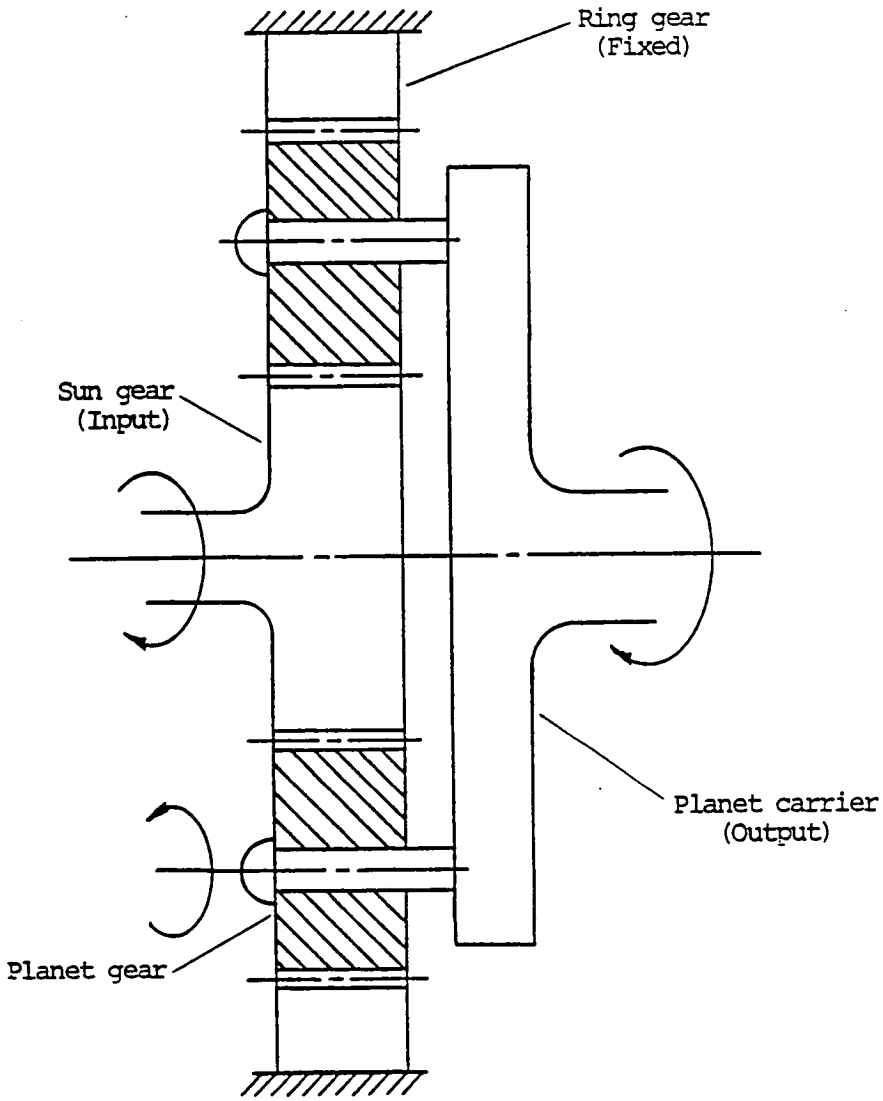


Figure 1 - Planetary Gear Train

Application	Turbo-pump		Turbo-generator	
Power	4360 h.p.		1060 kw.	
Speeds, rpm	6160/1120		6000/600	
Ratio	5.5 : 1		10 : 1	
Drive train type	Parallel shaft	Epicyclic	Parallel shaft	Epicyclic
Weight, lb.	3400	2500	3800	2200
Heaviest component, lb.	1730	1600	2100	1500
Circular pitch, in.	0.907	0.605	0.725	0.518
Pitch circle dia., in.	Pinion 5.774 Gear 31.754	Sun 4.619 Planet 8.083 Ring 20.785	Pinion 3.695 Gear 36.950	Sun 2.474 Planet 9.897 Ring 22.269
Pitch line velocity, fps	155	102	96.5	58.5
Horsepower losses				
Bearing	46.2	38	21	8.2
Tooth	26	21.4	9	8.1
Gearcase size, in.				
Length	42	36	37	30
Width	48.5	33	47	33
Height	36	32	48	32

Table 1 - Comparison of Parallel Shaft -vs- Epicyclic Gear Trains

that the use of multiple planets allows the load to be transmitted by several tooth contacts, and the co-axial arrangement of input and output shafts gives a more compact layout. For a planetary arrangement with three planets, each tooth engagement of the sun gear would have to carry one third of the total load. Consequently, the dimensions of the sun gear would be one-third of the pinion of a parallel shaft gear-train designed to transmit an equivalent torque.

It should be pointed out that six planets do not necessarily give a gearing twice the capacity of a similar one having only three planets. It is impossible to guarantee equal loading among more than three planets due to quality and accuracy of workmanship. For this reason, three planet designs are preferred, although systems with up to eight planets are commercially available.

A second advantage of using multiple planets is that when two or more planets are spaced symmetrically on the carrier, the radial loads of the planets offset each other. Therefore, the bearings and the gear housing for the co-axial elements must be designed only to maintain proper alignment of the gearing and withstand loads imposed by external conditions.

Another advantage of epicyclic gearing is that the smaller gears used can be made more accurately and with less difficulty than the larger ones of parallel shaft gearing. They are easier to handle and to harden, and distortion during hardening is not such a serious problem.

While the ring gear is not carburized because of the difficulty of precision grinding of internal teeth, it can be hardened by nitriding. This causes so little distortion that gears can be run without any post-nitriding processing. Normally, hardening of the ring teeth is not that critical since the surface stress between internal and external teeth is less than that between two external teeth. The concave surface of the internal tooth in contact with the convex surface of the planet tooth results in a larger contact area than for two external teeth, thus increasing the limiting wear load. Also, for a given size and number, teeth cut on an internal ring are stronger than those cut on an equivalent external wheel.

Use of smaller components gives lower pitch line velocities. This accounts for epicyclic gear trains being more quiet than parallel shaft gear trains. Having more teeth in mesh, not shifting the load so abruptly, also reduces the noise level.

Generally, single-stage epicyclic gear trains are more efficient than equivalent parallel shaft gear trains because power losses occurring through tooth friction and bearing losses are reduced. Tooth friction losses are approximately proportional to the tooth load and the pitch line velocities. With smaller tooth loads and slower pitch line velocities, the friction loss in epicyclic gears is less than parallel shaft gears running at the same rotational speed with the same load. Bearing losses are dependent on bearing size,

which are smaller on epicyclic gears since no tooth reaction loads are carried.

Epicyclic gear systems have a long history of industrial use. As early as 1781, James Watt patented a sun and planet gear arrangement used in one of his early engines. However, advances in internal gear manufacturing did not parallel those in external gears, limiting the development of epicyclic gear trains. As industrial applications required transmissions with higher power ratings, the performance of epicyclic gear systems became poorer at higher loads since load equalization among the planet gears was not realized due to poor manufacturing and assembly techniques.

However, the compact layout and inline arrangement found favor with the early automobile designers. Weight and size became important considerations since the power source and transmission were no longer stationary objects, but an integral part of the moving machine. Dr. F. W. Lanchester is generally credited with being the first to use epicyclic gears in automotive applications with the annulus of the first stage used as the planet carrier of the second stage to form a compound planetary transmission.

W. G. Stoeckicht adapted epicyclic gears for aircraft and marine applications. He designed the 3300 HP, 3200 to 1700 RPM gears for the Jumo 222, the largest piston type aircraft engine developed in Germany. He also designed a 5000 HP, 3770 to 550 RPM marine main propulsion unit, which had a gear case three feet in diameter, 2 1/2 feet long, and

whose total weight was less than one ton.

Presently, planetary gears are frequently used as main reduction gears in propulsion gas turbines for merchant ships. They are widely used in rotor drive gearboxes for helicopter aircraft. In lower horsepower applications, high ratio planetary systems are combined with hydrostatic drives to produce wheel drives for agricultural and off-highway equipment.

In an attempt to save weight and material and obtain better load sharing, current gear train designs are increasingly incorporating lighter, more flexible structures. As a result, at higher tooth-meshing frequencies, the dynamic behavior of the gears becomes of increasing interest because of its affect on gearbox noise and life and power rating of the transmission.

Efforts have been made to determine the instantaneous loads on the gear teeth, but few have examined the influence of the entire gear train on the dynamic loads. Most models used to date did not account for non-conjugate gear action caused by the deflection of teeth and other elements under load, or by inherent errors caused by gear manufacturing and assembly. Also, most models did not consider changes in the load sharing among planets when the number of engaged gear teeth at each planet change. Accurate modeling of the sun/planet and planet/ring tooth engagements directly affects the determination of the instantaneous load.

In addition to the tooth engagement variations, the

dynamic loading on the gear teeth is dependent upon the interaction of the components that make up the gear train and the load transmitted by the gear train. Previous investigators, for the most part, have not considered load variations which normally occur because of the elastic nature of the gear train elements. Usually the analysis is made considering constant torques applied to the input member, constant velocity ratios maintained between meshing gears, and a constant torque withdrawn from the output member.

More recently, large-scale digital computer programs have made it possible to investigate gear tooth interactions to permit a more realistic model for the epicyclic gear train. This would allow the study of dynamic loading by considering tooth engagement which is affected by characteristics of the gears in mesh, such as errors, tooth stiffnesses, masses of the gears and contact ratios. Also to be considered would be the influence of the gear train composed of connecting shafts, gears, bearing supports, couplings, and torque inputs.

It is the purpose of this investigation to develop a more comprehensive model which considers the affect of tooth engagement and system parameters. This model will be used to improve the current analytical methods for determining the instantaneous loads to which gear teeth are subjected. Since planetary gear systems are most commonly used, and techniques exist to determine equivalent planetary gear trains, a comprehensive method for analyzing the static and dynamic

loading in a planetary gear train will be developed using a variable-variable mesh stiffness (VVMS) model for external and internal gear teeth. No current technique uses a non-linear tooth mesh stiffness in determining planet load sharing, nor have any investigations been undertaken examining the effect of the phase relationships of the VVMS on the dynamic behavior of the gears. Consequently, this work extends the scope of engineering analysis of epicyclic gearing. The analysis is applicable toward either involute or modified spur gearing.

CHAPTER II

LITERATURE REVIEW

Since the planetary gear train (PGT) is an assembly of both external and internal spur gears, this literature review relies on information for PGT's as well as its individual components in showing the progress and current status of epicyclic gearing. Investigation into the action of the separate elements must be made in order to better understand the behavior of the entire system. Consequently, information is presented chronologically, and in separate sections, for external/internal spur gears and PGT's, respectively.

2.1 Spur Gears

A major factor in the design of spur gears is the power that must be transmitted from the primemover to the load. The force that is transmitted becomes important in designing for gear tooth beam strength, contact stress and scoring factor. Dynamic effects must be considered since the load is being transmitted by an elastic medium, i.e., deflected gear teeth. Consequently, the instantaneous load to which the engaged gear teeth are subjected is generally higher than the nominal statically calculated load.

During the late 1920's and early 1930's, the American Society of Mechanical Engineers Research Committee investi-

gated dynamic loading of gear teeth. Lewis and Buckingham conducted tests to determine the effects of operating speed and manufacturing errors in the involute tooth profile. Their report presented a method to evaluate the dynamic load increment due to gear train dynamics and tooth error. These studies presented a dynamic load solution more commonly known as Buckingham's Equation's [1]*.

Tuplin [2] used an equivalent spring-mass system representing the gears in mesh to determine the dynamic loads in gear teeth. The mass was determined from equivalent masses of the gears concentrated at the pitch circles. The spring stiffness was determined from the static load-deflection of two contacting teeth. It was considered constant and linear. He felt that dynamic loads occured from the passage of "thick" teeth through the meshing zone. The excess width impressed a displacement upon the mesh spring, introducing dynamic effects. Tuplin investigated different shapes of excitation, i.e., profile errors, and concluded that the shape of excitation had little effect on his results.

His calculated load increment was independent of the average transmitted load and was inversely proportional to the pitch-line velocity. Also, his equations did not account for system damping nor multiple tooth contact. In general, Tuplin's analysis can be considered as using a constant equivalent mesh stiffness.

* Numbers in brackets refer to bibliography

Bollinger [3] was one of the first to consider the tooth stiffness as a periodic function. Discontinuities in the effective stiffness between gear pairs resulted from the change from single to double tooth contact and the change back from double to single tooth contact. Results of his study correlated the experimental and analytical work very well.

This technique can be defined as a fixed-variable gear mesh stiffness model (FVMS). Although it is an improvement over the constant mesh stiffness model, the simplifications used in the model can be generalized as:

a. It does not account for irregular sharing of the load between simultaneously engaged tooth pairs. Therefore, analysis is limited to contact ratios below 2.0.

b. Gear tooth errors have negligible effect or none on mesh stiffness. This implies for a given load, a gear with errors will have an equivalent mesh stiffness as the same gear without errors.

c. Contact is assumed to occur only on the line of action.

d. The contact ratio and/or mesh stiffness is not affected by pre-mature or post-mature contact caused by deflection of the gear teeth under load.

e. The mesh stiffness has a fixed functional relation to the displacement of the gear.

In an effort to determine tooth mesh stiffnesses, investigations have been made to determine gear tooth deflections as a function of the point of load of application. Weber [4] used the strain energy technique along with the actual shape of the tooth profile in his deflection

analysis. Normal, shear, and bending energy in the tooth and its foundation was considered to the tooth deflection. Hertzian deformation was calculated using the tooth profile radii of curvature as equivalent cylinders. Attia [5] expanded Weber's model by including circumferential deformation of the gear rim and the deflection of a tooth under the effect of loaded neighboring teeth.

Kasuba and Evans [6] used a large scale digitized approach to calculate directly the gear mesh stiffness of two external spur gears as a function of transmitted load, gear profile errors, gear tooth deflections, gear hub deformations, and position of tooth contact and the number of tooth pairs in contact. By introducing these aspects, the calculated gear mesh stiffness (Fig. 2) was defined as being a variable-variable mesh stiffness (VVMS). The VVMS model has the following properties:

a. It simulates the gear system by including the elastic effects of the entire system and not just the gears.

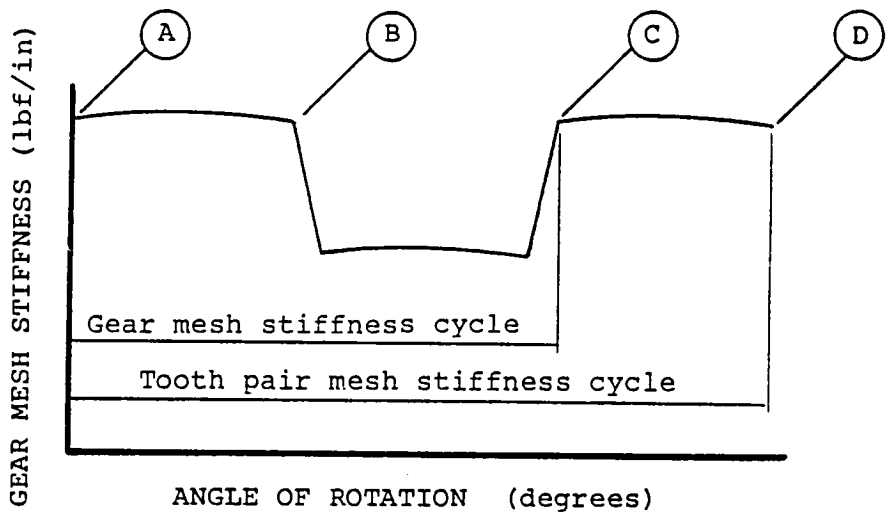
b. The stiffness of the gear teeth is considered a function of the position of contact.

c. It allows for multiple tooth contact by examining each pair of teeth that are in and close to the theoretical contact zone.

d. The gear tooth profiles are defined with errors to simulate tooth profile errors found in manufacture.

e. It allows for backlash in the gears.

The actual load sharing and deflection are calculated for discrete positions within an established mesh arc. For



Point A - Reference tooth pair enters contact zone
Point B - Preceding tooth pair exits contact zone
Point C - Following tooth pair enters contact zone
Point D - Reference tooth pair exits contact zone

Figure 2 - Variable-Variable Mesh Stiffness

any position i, the calculated k^{th} gear tooth pair stiffness $KP(k)_i$, combined mesh stiffness KG_i, and load sharing incorporate the effects due to profile errors, profile modifications, and tooth deflections. The individual gear pair stiffness then is

$$KP(k)_i = Q(k)_i / \delta(k)_i \qquad (2.1)$$

where: $Q(k)_i$ = gear pair static tooth load

$\delta(k)_i$ = tooth deflection due to bending, compression, and shear forces; and total hub deformation

If the effective errors prevent contact, $KP(k)_i = 0$.

The mesh stiffness at the i^{th} position is the sum of the gear tooth pair stiffnesses for all pairs in contact at position i,

$$KG_i = \Sigma KP(k)_i \ . \qquad (2.2)$$

The concept of the VVMS was further expanded by introducing an iterative procedure to calculate the VVMS by solving the statically indeterminate problem of multi-pair contacts, changes in the contact ratio, and mesh deflections. The results of the VVMS model showed that:

a. The maximum instantaneous load occurred immediately after a change in the number of teeth in contact.

b. Dynamic load factors decrease with increasing average transmitted load between gears.

c. Dynamic load factors vary with the speed of operation of the system and closeness to the resonances of the system.

d. The gear system can be tuned through the use of torsionally flexible gear bodies/hubs/rims to reduce dynamic load factors.

e. Dynamic load factors are reduced when the contact ratio is increased.

f. Dynamic load factors are reduced when the system damping is increased.

Pintz [7] used a similar technique for developing the VVMS for an internal-external gear mesh. This was used to determine the dynamic loads experienced in an internal gear drive. Pintz's model used an internal ring gear which was radially supported and driven by an externally tooth gear. He stated that the VVMS model was appropriate in light of the "very high contact ratios" encountered with internal gear drives. Use of the VVMS allowed investigation into problems unique to internal gears such as ring gear deflections. He examined the effect of ring deflections on the gear mesh, and found considerable missing and backhitting of the gear teeth similar to the performance of gears with sinusoidal profile errors. Also, larger dynamic load factors were found with an increase in radial deflection.

The work of Kasuba and Pintz represents the most effective way to date of determining the discontinuous, non-constant, gear mesh stiffness.

2.2 Epicyclic Gears

Epicyclic gears have been used as early as the 1700's. However, the first recorded attempt to apply engineering analysis was made in the very late 1800's. Lanchester [8]

recognized the need for "precision in workmanship" to achieve optimum performance from the planetary gear train by equal load distribution among the planet gears. He found that the working clearance for a three planet system allowed the planets to find an equilibrium postion such that the load was evenly distributed among the planets. But for a four pinion combination assembled with the highest degree of accuracy possible, it was found that the load was not quite so evenly distributed among the four planets.

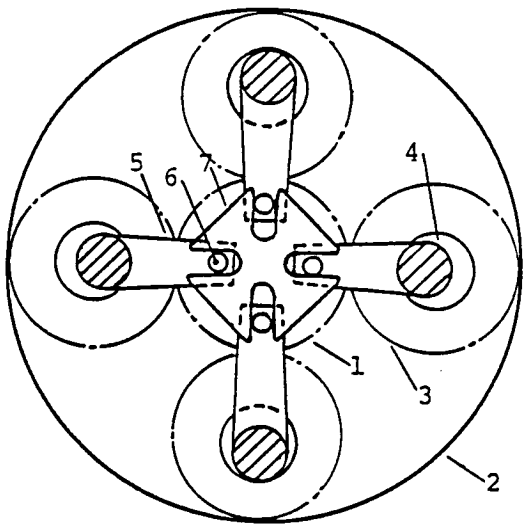
Wilson [9] was able to show that his three speed epicyclic gearbox achieved an efficiency of 99.25%, which was a marked improvement over existing parallel shaft, sliding-gear transmissions. He attributed this to reduced oil churning in the transmission. The centripetal force of the rotating components forced out any surplus oil, leaving just the necessary oil film to lubricate the gears.

Love [10] described a two stage design process for epicyclic gearing. The first is the determination of relative dimensions based on gear ratios required and the physical constraints imposed by assembling an epicyclic gear such that the planets are symmetrically placed about the sun. The second stage is the determination of absolute dimension based on considerations of power to be transmitted, running speeds, and the strength of available material of construction.

Seager [11] proposed an approximately theoretical analysis of loading in a simple planetary gear system for

both centered and non-centered (floating) gears. His analysis assumed errors in the planet positioning and planet tooth thickness, and a constant combined tooth stiffness independent of meshing position. Inertia forces were neglected, limiting the theory to low speed behavior with minimal dynamic effects. Based on the equilibrium conditions about the sun gear and compatible displacements among the planet gears, Seager developed a set of equations to determine load sharing among the planet gears. It was shown that for three planets with a floating sun gear, the total load is shared equally among the three planets, irrespective of the distribution of errors. However, it was found if the number of planets was greater than three, the load was no longer divided equally among the planets. Furthermore, the maximum planet load depended on whether the planet lagged or lead its ideal position.

Imwalle [12] classified designs to achieve load equalization among planets in two systems, statically indeterminate and statically determinate. Statically indeterminate systems are ones which rely on precise tolerances of the manufactured componests to minimize load imbalances. These systems do not lend themselves to rigorous mathematical analysis. Statically determinate systems accomplish load equalization by introducing additional degrees of freedom into the system. This is done by the use of floating members or by allowing rigidly connected members to adjust their position to provide equalizing action (Fig. 3). Imwalle also developed a



1 - Sun gear
2 - Ring gear
3 - Planet gears
4 - Eccentric pivots
5 - Arms
6 - Rollers
7 - Maltese-cross-type equalizing member

Figure 3 - PGT with Floating Members

method for comparing different load equalization systems from kinematical and dynamic considerations. He assumed the total dynamic force to be the sum of two components,

$$F_d = F_d' + F_d'', \tag{2.3}$$

where F_d' = Dynamic force independent of load equalization system and used to analyze non-uniform rotary motion of the principle parts.

F_d'' = Dynamic force dependent upon the type of equalization system and its masses.

The F_d'' component can be used as a design parameter to evaluate the effect of the system on the total dynamic force. In general, the lowest value of the dynamic force for a given system provides the best load equaliztion. Imwalle's analysis assumes the planetary gears are undamped and neglects the mesh stiffness and ring gear flexibility. Consequently, the analysis does not account for vibratory phenomena in the gear train.

Seager [13], noting that even with high quality gears, vibration and excessive noise is experienced; tried other means to reduce vibration in epicyclic gearing. He showed that some potentially troublesome harmonic components can be neutralized by suitable choice of tooth numbers. Seager stated that the primary source of tooth excitation is static transmission error, whose fundamental frequency is the mesh frequency. The possibility of neutralizing excitation by the teeth in epicyclic gearing comes from the fact that, in general, the planets are at different phases of the tooth

meshing cycle. When the torsional and transverse modes are effectively de-coupled, the conditions for neutralizing the components of excitation are:

$$\frac{Z_s}{n} \neq \text{integer} \quad \text{(torsional modes)} \tag{2.4}$$

$$\frac{Z_s \pm 1}{n} \neq \text{integer} \quad \text{(transverse modes)} \tag{2.5}$$

where: Z_s = Number of teeth, sun gear

n = Number of planets

These conditions can be satisfied provided that the number of planets is greater than three, and are only valid if all the planets are identical and there are no errors in their position.

Cunliffe and Welbourne [14] built and tested a single stage epicyclic gear train in order to see whether agreement could be obtained between the model of it, and dynamic force measurements make on the gear. Their model had 13 degrees of freedom and simulated a star gear system with a stationary carrier and rotating annulus. Their experimental results show that except for very highly loaded, very accurately made gears, uniform load sharing cannot be expected even under quasi-static conditions, i.e., tooth mesh frequencies below the overall fundamental frequency. Load distribution below the fundamental frequency can be improved by reducing any of the component stiffnessess. Above the fundamental frequency,

the system behaves dynamically. The general results imply the possibility of dynamic tooth loads of three or four times mean tooth loads at low frequency resonances, and up to four to six times for high frequency resonances. The thirteen mode shapes obtained with the model were classified as to low or high frequency modes. The seven low frequency modes were strongly influenced by support stiffness in their dynamic behavior. Maximum tooth load could be varied by altering planet pin stiffness to the mesh stiffness ratio. The six high frequency modes were little affected by planet pin stiffness. They were, however, critically dependent both on tooth contact stiffness and accuracy of manufacture. The excitation of particular modes could be reduced by the appropriate choice of symmetrical or asymmetrical meshing of the planets through selection of tooth numbers in the gear.

In their investigation of noise generated by planetary ring gears in helicopter aircraft transmissions, Chiang and Badgley [15] studied the mechanism by which gear meshes generate vibrations. These vibrations, which result from the deviation from uniform gear rotation, are one of the major causes of gearbox noise. Using known dynamic forces as inputs in a forced vibration analysis, Chiang and Badgley were able to determine ring gear vibration mode shapes and amplitudes. These parameters were used to measure the effectiveness of design modifications in reducing vibration amplitudes and decreasing dynamic forces in the gear train. Their analysis also took into account the phase relationships

of the dynamic forces on each planetary gear. That is, when one of the planet gears reaches its peak dynamic force, the remaining planet gears are at intermediate forces below the peak value. They concluded that structural modification to the ring gear to separate natural frequencies from multiples of lower mesh frequency would be most beneficial from the standpoint of ring gear vibrations.

Tucker [16] studied the effects of flexible ring gears in epicyclic gear trains. He concluded that rings with internal gear teeth have substantially different deflection characteristics than discs with external gear teeth that will affect the gear train performance. The greater resilience of the ring gears, results in greater deflections in the internal teeth than the external teeth, which have a more rigid support. These deflections in turn, affect the tooth stiffness, which is a major factor in determining the dynamic loads on the teeth. The dynamic load increment, which would be added to the transmitted load, would be:

$$W_d = \frac{k_r e}{\left[\frac{k_r t^2}{4M} + 1\right]^{\frac{1}{2}}} \qquad (2.5)$$

where: W_d = Dynamic load

k_r = Effective tooth stiffness on ring gear

e = Effective tooth spacing error

t = Tooth contact time

M = Equivalent mass of two mating gears

The effective stiffness of the ring gear tooth is:

$$k_r = \frac{W_t}{\Delta + \Delta'} \qquad (2.7)$$

where: W_t = Tangential load on tooth

Δ = Lateral displacement away form mating tooth due to radial deflection

Δ' = Lateral displacement away from mating tooth due to bending moment

Tucker observed that the flexibility of the ring is so much greater than the tooth that the ring spring rate governs the dynamic load increment of the tooth. Therefore, at high speeds, a ring gear generates much lower dynamic loads than an external gear with equivalent tooth errors.

Tucker felt an important consideration during the design phase of an eqicyclic gear set was the calculation of the natural frequencies of the ring gear. These frequencies may be excited by forcing functions generated during the operation of the gear set and result in fatigue failures. The forcing functions which can generate such frequencies are, in ascending order:

a. Ring gear teeth spacing errors
b. Planet gear teeth spacing errors
c. Oval ring gear
d. Oval planet gear
e. Sun gear tooth errors

f. Oval sun gear

g. Tooth meshing frequency

A frequency interference (Campbell) diagram should be drawn to determine operating speeds where resonance could be expected, since the variation of the forcing functions is a linear function of the input gear speed (Fig. 4). If a forcing function frequency is coincident or close to a ring natural frequency, the ring thickness should be changed to avoid problems.

Nakahara, Takahasi, and Makuso [17] examined the practice of using an intermediate ring between the planet gear and planet gear pin to improve load sharing among planets. They point out that a floating sun gear is not very effective in equalizing planet loads if the system has four or more planets. Flexible ring gears cannot often be used to offset errors because resonance conditions may occur due to the decrease of natural frequencies. However, inserting an intermediate ring (IR) between the planet gear and its spindle has been used successfully to eliminate the influence of relatively large errors on the load distribution among planets. In general, the effectiveness of the intermediate rings is due to increasing the compliance related with the distance between the planet gear and spindle center independently of the other planet loads. Two different types of intermediate rings are used, depending on the application (Fig. 5).

Figure 5a is used for the case of relatively low speed

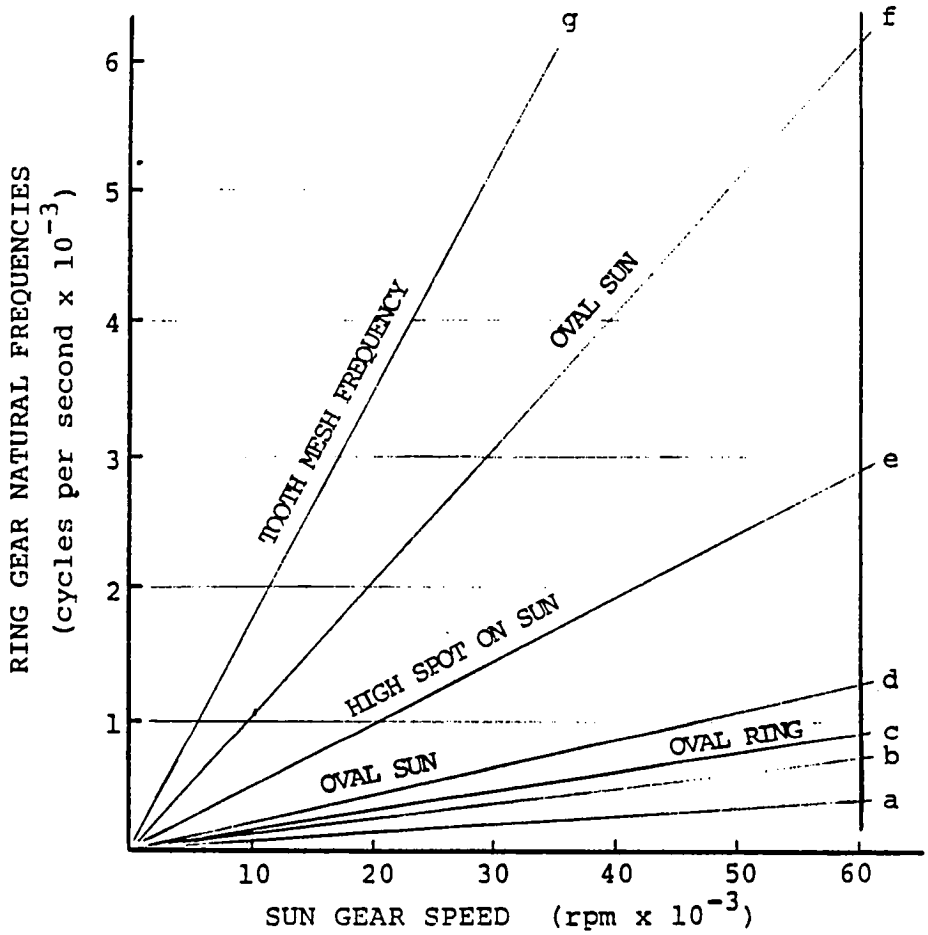
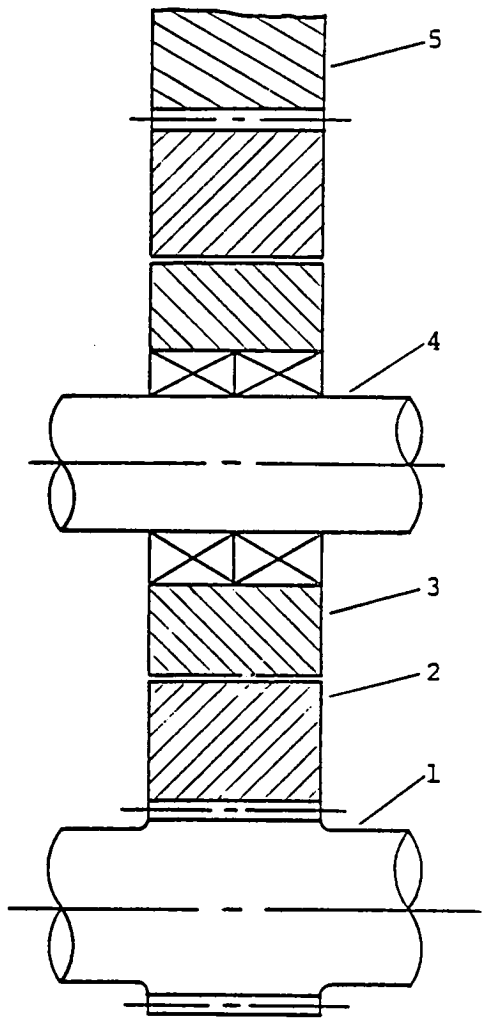


Figure 4 - Campbell Interference Diagram



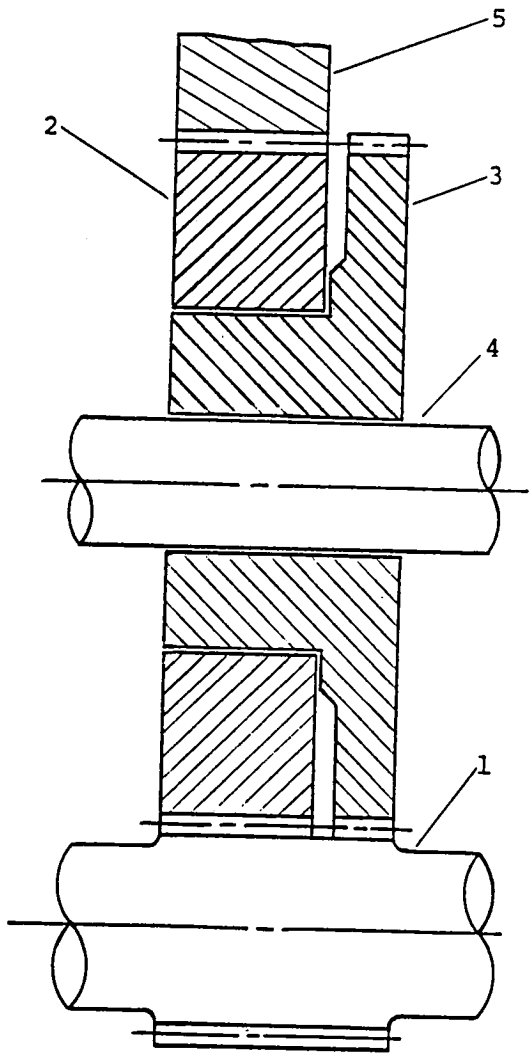
1 - Sun gear
2 - Planet gear
3 - Intermediate ring
4 - Planet gear spindle
5 - Ring gear

Figure 5a - Low Speed Intermediate Ring Structure

and high torque. Here the IR can rotate at almost the same angular velocity as the planet gear without help of forced driving. The IR serves to distribute the planet load to pin roller bearing over a wider arc while maintaining a concentrated load on the ring gear. The resulting elastic deflections of the ring gear then act to offset the system errors. The reduced Hertzian contact stress also helps to prolong bearing life.

Figure 5b is used for the case of high power, high speed applications, such as marine engine transmissions. In this case, it is necessary for the IR to be driven by the sun and internal gear in order to rotate with the planet. The ring is called a "floating" intermediate ring because it floats on an oil film between the planet gear and spindle. The relatively thick oil film in the clearance improves the compliance of the planet. With high centripetal forces due to high carrier angular velocities, the lubrication film also increases the damping characteristics significantly.

Botman [18] evaluated the various vibration modes of a single stage planetary gear in order to predict the gear tooth loads. Figure 6 shows the dynamic model for only one planet. Each component has three rigid body degrees of freedom. The planets are symmetrically spaced about the sun and attached to the carrier by means of two equal springs attached in two perpendicular directions. The total number of degrees of freedon was 9+3n, where n is the number of planets. The tooth mesh stiffnesses were assumed to be



1 - Sun gear
2 - Planet gear
3 - Intermediate ring
4 - Planet gear spindle
5 - Ring gear

Figure 5b - High Speed Intermediate Ring Structure

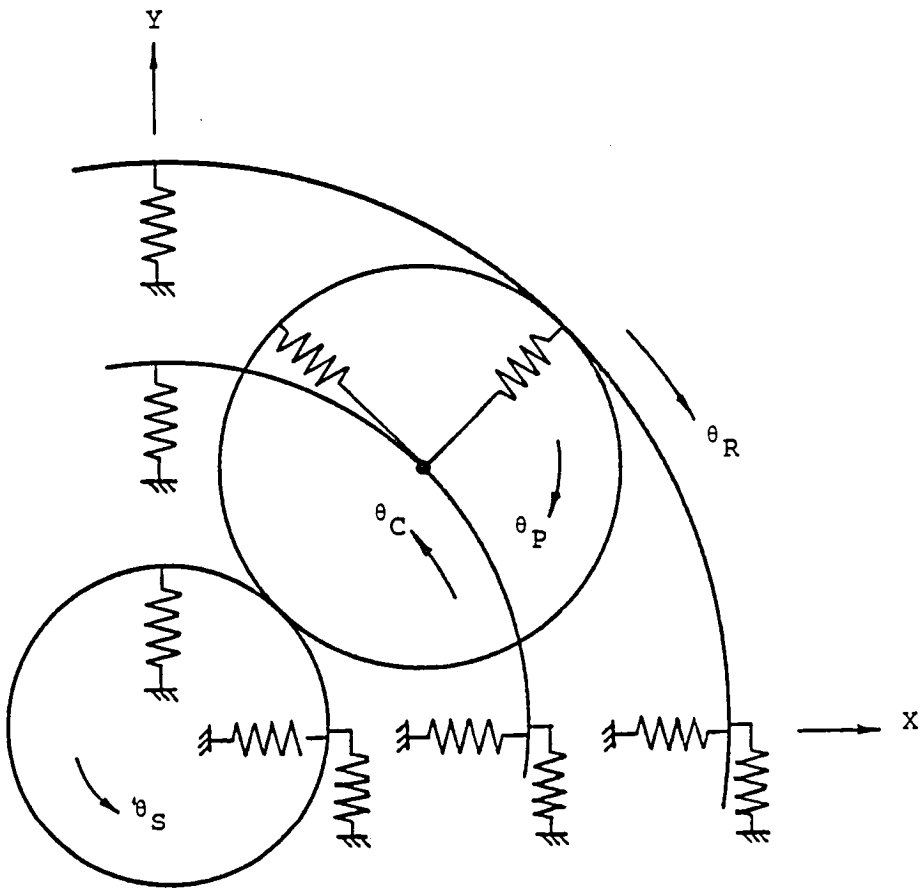


Figure 6 - Botman's Dynamic Model (One Planet Shown)

constant. The equations of motion in the most general form are:

$$[M]\{\ddot{q}\} + [C]\{\dot{q}\} + [K]\{q\} = \{F\} \tag{2.8}$$

where q represents the generalized coordinates, and M, C, and K are the inertia, damping, and stiffness matrices, respectively, for the components.

Vibration modes obtained for free vibrations with a non-rotating carrier were categorized into two groups, axisymmetric or non-axisymmetric. In the axisymmetric, or rotational, modes, all the planets perform the same motion with respect to the sun, and the other components have only rotational vibration. There were six such modes. The remaining twelve modes were non-axisymmetric. The planets did not all perform the same motion and at least some of the other components had lateral motion. The effect of carrier rotation on natural frequency was found to be small. The maximum effect was a reduction in the natural frequency of the lowest mode of about 8 percent at design speed.

Only the natural frequencies corresponding to the six rotational modes were found. Hence, Botman concluded that the lateral modes are suppressed by carrier motion. He also felt that the dominance of rotational modes suggested it was more meaningful to measure accelerations of the ring gear in the tangential, rather than radial direction. Since gear mesh frequencies tend to be high, the high frequency rotational modes were found to be the significant contributing

factor in the dynamic tooth loads. Influence of torsional input or output shafts would occur only if components also had a high natural frequency.

Hidaka, et al. [19, 20, 21, 22, 23], did a comprehensive study of the dynamic behavior of planetary gears. Using a Stoeckicht Type 2K-H, single stage planetary gear system with spur gears, Hidaka performed experimental studies to determine its dynamic behavior and how it is influenced by the operational characteristics of the planetary gear train, such as operational load, variation of torque, vibration characteristics of the system, and errors of manufacture and assembly. In conjunction with the experimental results, Hidaka attempted, analytically, to calculate the load distribution among planets, displacement of floating members, and dynamic tooth loads.

Hidaka first studied the problem of load distribution among the planets. The Stoeckicht planetary gear train used had both a floating sun and ring gear to eliminate the influence of manufacturing errors. Fillet strains were measured 30 times during the interval of one synchronous position to the next of the gear train. Tooth loads were then measured from these fillet strains. The dynamic load was defined as the ratio of dynamic fillet strain to static fillet strain. The load distribution among the planets was then found from the distribution of fillet strains on the sun gear teeth. In order to estimate the load distribution and dynamic tooth load from the measured fillet strains, Hidaka

calculated the mean dynamic load factors and the standard deviation through one synchronous interval. A coefficient of variation was defined as the ratio of standard deviation to the mean value. It was found that for slow input speeds, the load was equally distributed among the three planets. For higher input speeds, the mean load distribution remained nearly equal for the planets. However, the sun gear teeth experienced increased dynamic loading with increased input speed. Hidaka concluded that this variation in dynamic load was largely due to torque variation. It also meant that the dynamic tooth load could not be predicted just from the mean load distribution and the mean dynamic load; the coefficient of variation for each of these parameters must also be calculated. Using the mean tooth loads and the coefficient of variation from the mean of those loads, Hidaka derived equations for both the dynamic tooth load and the load distribution among the planets.

Hidaka found that the dynamic tooth load and the load distribution among planets is closely related with the displacements of the sun and ring gear. He found that in the low speed region, displacement of the sun gear served to equalize the planet loading. The sun gear moved on the locus fixed by the influences of the random errors of manufacture and assembly and the elastic deformation of the components. At higher speeds, the amplitude of the displacement of the sun gear increased and its motion became more circular. Generally, the periodicity of the sun gear displacement was

comprised of the following three components:

a. A component of frequency coinciding with the mesh frequency, representing the stiffness function.

b. A component of frequency coinciding with the frequency of revolution of the input shaft, representing run-out errors.

c. A component of frequency coinciding with the frequency of revolution times the number of planets, representing errors of assembly and deformation of the ring gear.

As the sun motion became more circular, the coefficient of variation for the dynamic load factor and the coefficient of variation for the load distribution rate both increased drastically. The larger displacements of the sun gear affected the meshing conditions between the sun gear and the planets so that dynamic loads were now being introduced on the gear teeth. Also, at higher speeds, resonance occured in the ring gear if the mesh frequency of the planet/ring approached the natural frequency of the ring.

Hidaka examined ring gear static displacements with finite elements to derive approxiamte equations for the calculation of deflections of internal gear teeth. The total deflection of the internal gear tooth was obtained by superimposing the deflections resulting from the bending moment, shearing force, and inclination at the root of the tooth upon half the deformation resulting from Hertzian contact. The deflection of a loaded tooth causes the adjacent non-loaded tooth to decrease the relative approach distance between teeth. Consequently, premature contact occurs and the

contact ratio is increased. Hidaka later determined that having a ring gear too thin produces the undesirable result of tip interference with an approaching tooth due to tooth deformation. He also found that except at resonance, the influence of the rim thickness of a ring gear on the dynamic tooth load is small.

Hidaka developed a dynamic model based on the characteristics obtained from the experimental results. Important assumptions made were:

a. Planet gear shafts were very stiff and deflection of these shafts due to the centripetal force of the carrier rotation were neglected.

b. Translation of the ring gear was negligible.

c. Sun gear displacement was restricted to planar motion, i.e., it did not wobble.

d. Tooth stiffnesses of the sun/planet and planet/ring meshes were assumed to be constant.

The rotating inertia and the torsional stiffness in the planetary gear train were transformed into the equivalent mass and stiffness along the line of action. The angular displacements of each component were then also transformed into linear displacements in the direction of the line of action. The equivalent masses moved up and down corresponding to the direction of the line of action, but did not wobble. Since it is possible for the planet gears to rotate freely about their shaft, the planet gears were treated as behaving like seesaws. The driving and load elements, respectively, were connected by flexible shafts to the sun

and ring gear.

The differential equations were set up by summing the forces on the elements and solved by means of the CSMP simulation program for continuous systems. The integration period was taken as ten revolution of the sun gear (140 pitches) and the dynamic tooth loads were calculated during 30 pitches.

Hidaka concluded that the calculated results satisfactorily agreed with the measured ones. Calculations of the dynamic load factors were found to be low at the higher speeds. Hidaka felt this was due to the fact that the main cause of the variation of the dynamic load was tooth errors and the tooth stiffness varying once per pitch; something he was not able to duplicate by assuming a constant average mesh stiffness. Calculations of the load distribution rates and variation showed tendencies similar to those of the measured values, but again the measured load variations were higher in the high speed range. Hidaka concluded that this implied that load unbalances caused by runout errors could be compensated by a floating sun gear. However, with increasing speeds, as the sun gear assumed a circular motion, the effects of tooth profile error and stiffness variation increasingly become prevalent factors in the dynamic loading.

Jarchow and Vonderschmidt (24) developed a model with three planet gears taking into account manufacturing and tooth errors and tooth stiffness variations which cause vibrations with subsequent inner dynamic tooth forces. A

system of coupled, non-homogenous, non-linear differential equations of motion were used to find the tooth forces in the planetary gears. Their study investigated the influence of transmitted load and input speed on the effective dynamic tooth force. Three configurations were evaluated:

a. All coaxial elements fixed

b. Sun gear allowed to float

c. Carrier allowed to float

Two parameters were defined in the load calculations:

a. Planet gear load distribution factor, K_γ, which takes into account the non-uniform load distribution on the individual planets (K_γ = 1.0 for ideal load distribution).

b. Dynamic load factor K_v, which considers the additional dynamic tooth forces.

It was found that for case a, the load distribution factor decreased with an increase of load, implying influence of manufacturing errors relative to static load decreased with increasing load. For cases b and c, K_γ was independent of the transmitted load.

The product, $K_\gamma K_v$, was defined to be the load magnification factor. Generally, it decreased with an increase of load. Case a had the highest load magnification factor, while Case c had the lowest. Jarchow's experimental work corroborated these findings. It was found that $K_\gamma K_v$ increased strongly with the increase of sun gear speed. One reason given was that the entry and exiting impulses increase with

the increased speed. Again, Case a and Case c had the highest and lowest load magnification factors, respectively.

CHAPTER III

ANALYTICAL INVESTIGATION

3.1 Problem Formulation

A complete analysis of the dynamic loading of a planetary gear train (PGT) is difficult even under ideal geometry conditions because of the non-singular engagement conditions at each sun/planet and planet/ring mesh. Additional difficulties arise because of variations in the load transmitted by the PGT. These variations are primarily caused by discontinuities in the mesh stiffness resulting from the continuous changing, depending on the contact ratio, from n to n+1 tooth contact and back to n tooth contact. Dynamic effects are introduced as the transmitted load is periodically shared between n and n+1 teeth. Further complications are introduced by transmission errors caused by gear manufacturing errors. A thin tooth or a pit in the profile may temporarily prevent or interrupt contact and cause sudden disengagement and subsequent clashing between teeth. The nonlinearity of the mesh stiffness is also greatly affected by the magnitude of the transmitted load. In some instances, with a large enough load, the compliance of the teeth is great enough so that the magnitude of the elastic deflection is greater than the manufacturing error, restoring any interrupted contact.

Since the variable mesh stiffness for any two gears produces parametric excitation, a PGT system with n planets will have 2n variable mesh stiffnesses influencing the system dynamic behavior.. Figure 7 shows a three planet PGT with input torque at the sun gear. Although the three planets are spaced symmetrically around the sun gear, the contact points, and therefore the mesh stiffness, vary at each engaged gear pair. Consequently, a phase relationship can be established for the mesh stiffnesses and the dynamic forces (Fig. 8). Physically, this phase variation is dependent on the position of the planet gears. Also, for a system with multiple degrees of freedom, instead of having well defined critical speeds, regions of critical speed occur within which large vibrations may occur [25].

In addition to the ring and mesh stiffness affecting the dynamic behavior, the sun support stiffness must also be considered. As pointed out in the literature survey, a common technique for improving load sharing among planets, is to allow the sun gear to translate and find a position of equilibrium loading. The magnitude of the sun motion will then be dependent on the force imbalance. At high input speeds, the dynamic effects of the sun's translation may become appreciable. This would be especially true if a forcing frequency occured near a basic mode of excitation frequency.

The non-linearities in the geared system give rise to instantaneous load fluctuations between gear teeth despite

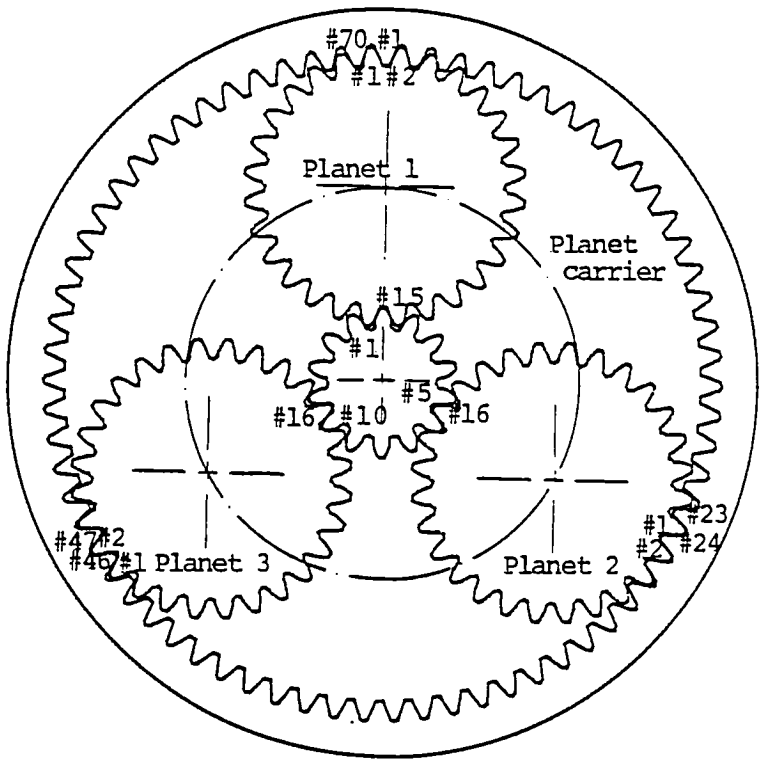
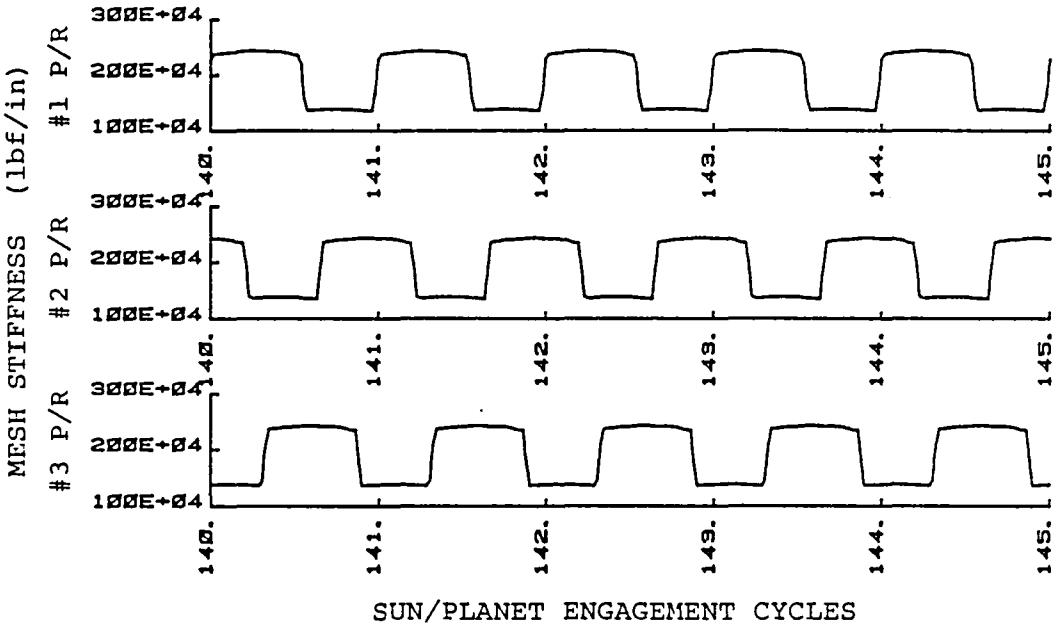


Figure 7 - Three Planet PGT



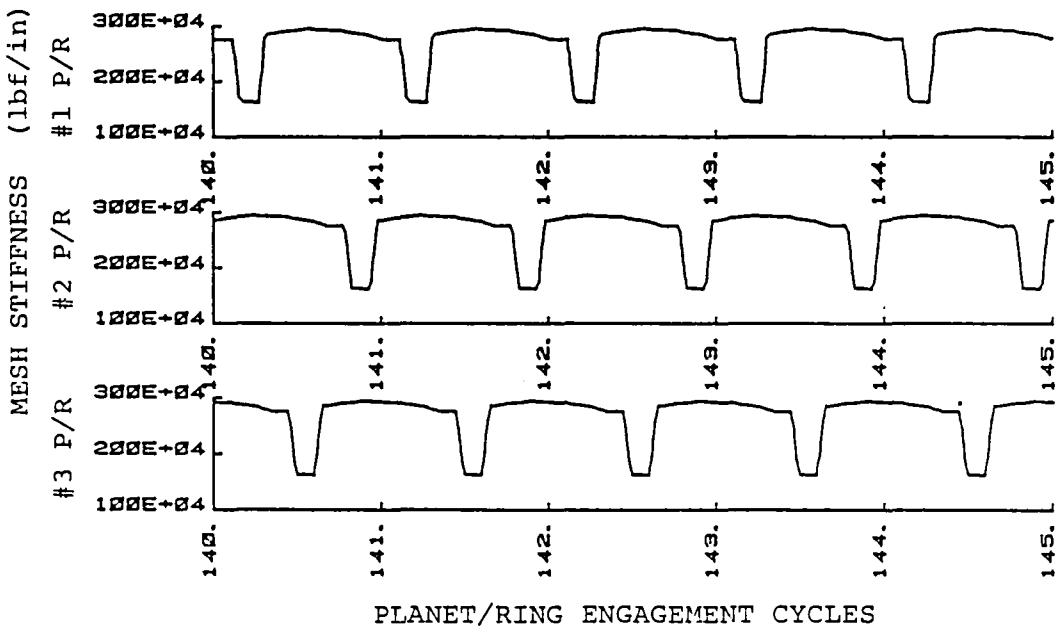


Figure 8 - Typical Phasing among Mesh Stiffnesses

apparently constant load conditions. Furthermore, the magnitude and duration of these fluctuations is influenced by the physical characteristics of the geared system. This would include the masses and stiffnesses of the components as well as damping and the proximity of natural frequencies with any of the forcing functions. Having stated the problem, this author feels that adequate mathematical tools and numerical techniques exist for their solution. The task remains to develop a mathematical model that accurately describes the physical system and provides meaningful results. Not only will new techniques have to be developed but improvements will be made to existing ones to provide a more efficient analysis.

3.2 Assumptions

The following assumptions and conventions are used to make the analysis manageable while maintaining a realistic model:

a. For static equilibrium conditions, two torque balances must be maintained:

   i. Collectively, the sun gear mesh forces must create a torque equivalent to the input torque.

   ii. The sum of the torques acting on each planet must be zero.

b. Since there is no net torque acting on the planet, there is no net planet hub deformation. Therefore, sun/planet and planet/ring deflections occur, and

may be treated, independently.

c. The analysis will be limited to a single planetary gear stage and to vibrations in its plane. Vibrations out of the plane of the gear stage are assumed negligible. This assumption is reasonable when tooth loads are confined to one plane, as they are in the spur gears being considered.

d. The differential equations of motion are expressed along the actual, rather than the theoretical line of action. This allows for evaluation of non-conjugate gear action due to transmission error.

e. The dynamic behavior of the PGT will be investigated through several complete engagement or mesh cycles to insure steady-state operating conditions.

f. The load-deflection behavior of the tooth pairs in mesh can be represented by a single non-linear spring. The gear mesh stiffness is determined as a function of transmitted load, gear tooth profile errors, and gear tooth and hub deformations. The component of dynamic transmitted load due to the spring force is proportional to the relative displacements of two gears along the instantaneous line of action.

g. Gear errors within a component are periodic and non-singular. Tooth deflections are localized and

centerline positions of gear teeth non-adjacent to loaded teeth remain rigidly fixed. As a tooth is being unloaded it regains its original shape and regains its rigid body position as the tooth immediately preceding it becomes unloaded.

h. Damping in each gear mesh due to lubrication is represented by a single dashpot assuming Newtonian damping. A constant damping coefficent of $\zeta = 0.1$ is chosen [26]. The critical damping ratio is defined as a function of the total effective mass and instantaneous stiffness. The component of the dynamic transmitted load due to viscous damping is proportional to the relative velocities of two gears along the instantaneous line of action.

i. The presence of backlash may lead to tooth separation under certain operating conditions. As such, the differential equations of motion must consider a period of no transmitted load before the contact is restored.

j. In order to isolate the vibratory phenomena within the gear train and minimize external influences, the input and output shafts are made very stiff torsionally. Masses of the driver and load are also much greater than that of any of the gears.

k. Displacement of the sun gear caused by allowing it

to float is restricted to planar motion, i.e., it does not wobble.

3.3 Method of Solution

The purpose of this investigation is to develop static and dynamic load analysis techniques for a planetary gear train. The methods presented in the literature review serve as the basis for the development of the analysis. The investigation is divided into static and dynamic sections, with the static results used as a baseline for comparison with the dynamic results. The determination of static load sharing among the planets will be used as the starting point.

An important task is the development of the mathematical model of the PGT. This allows continuous tracking of the individual gear components, the forcing function, i.e., driving torque, and the load. An accurate determination of the position of the elements is critical since the mesh stiffnesses will be considered as functions of gear displacement. The model also takes into consideration:

a. Floating or fixed sun gear
b. Gear quality
c. Positioning errors of components

For a PGT with n planets, the mathematical model has n+4 masses and, depending on the fixity of the gears, up to n+6 degrees of freedom. The model is used to determine the extent of the dynamic load increment and its dependence on

component stiffness, input speed, transmitted load, and planet load sharing. It is also used to study the sun gear motion to determine how tooth profile errors affect the sun/planet mesh, and the effect of the sun's transverse motion on planet load sharing. This model has been incorporated into a set of computer programs so that the static and dynamic behavior of the individual components, as well as that of the complete gear train, can be examined.

The programs have been developed for use on a minicomputer system such as the Hewlett Packard HP-1000 at Cleveland State University (CSU). They are incorporated into a computer-aided-design package that will be comprised of the analyses programs as well as menu-driven, interactive pre- and post- processors. This design package will enable the designer/analyst to evaluate a PGT design during one relatively short terminal session, instead of the rather lengthy time involved with a mainframe-supported batch system. The inter-active pre-processor allows for easy changes in the input data, while the post-processor offers the option of either tabular or graphical output for the user selected results.

The analyses programs themselves were developed keeping in mind the memory partition limitations of minicomputers. For example, the CSU mainframe IBM 370/158 has 4 megabytes of main memory, while the HP-1000 Model F contains 512K bytes of main memory. Efficient algorithms were developed as well as computer techniques, so as not to exceed the limitations of

the computer. The computer program was used to conduct parametric studies to determine the effect of manufacturing errors, tooth profile modifications, and errors damping, component stiffness, and transmitted torque on dynamic behavior.

The next three sections of this thesis present a detailed development of the static and dynamic analysis as well as a description of the computer program. For reference purposes, Appendix A explains the techniques used to evaluate the individual static gear mesh behavior. Appendix B shows how the methodology developed here can be extended to include the star and solar type of arrangements of epicyclic gearing. Appendices C and D contain the computer program listings.

3.4 Static Analysis

As mentioned in the previous section, the strategy for solution includes investigating the static loading conditions in preparation for the dynamic analysis. The accurate determination of the inter-dependence of the components' variable mesh stiffnesses and planet load sharing is of critical importance to successfully analyse the dynamic behavior of the PGT. The tasks of the static analysis portion of the investigation are:

a. Determine an equivalent gear train with stationary carrier.

b. Determine phase relationships of all gear mesh stiffnesses, including the effects of positioning errors.

c. Determine a synchronous position as a function of

the sun gear rotation.

d. Examine sun gear position and planet load sharing as a function of the sun gear rotation.

In addition, the results of the static analysis must be stored for use in the dynamic analysis and for printing of selected portions of this material.

3.4.1 Equivalent Gear Drives

In a PGT, the power circulating within the system components differs from the actual power being transmitted. This results from the relative motion of the gears and is present in some degree in all planetary gear trains. Radzimovsky [27] described a method by which an equivalent conventional gear train can be developed in conjunction with the particular kinematic characteristics of the planetary system. The equivalent conventional gear train would be one where the transmitted forces remain the same but the gears would only rotate about their own center, simplifying the dynamic analysis.

In the conventional gear train (Fig. 9), the pitch line velocities of the two gears are the same and the power output by the driven gear is equal to the power input to that gear. Figure 10 shows the meshing relations in a PGT. The velocity of engagement of the sun/planet and planet/ring gear is affected by the relative motion of the planet carrier. For the planet gears, the velocity of engagement, v_e, is not equal to the pitch line velocity, v. If F is the tangential force acting at the pitch circle of the sun gear, the product

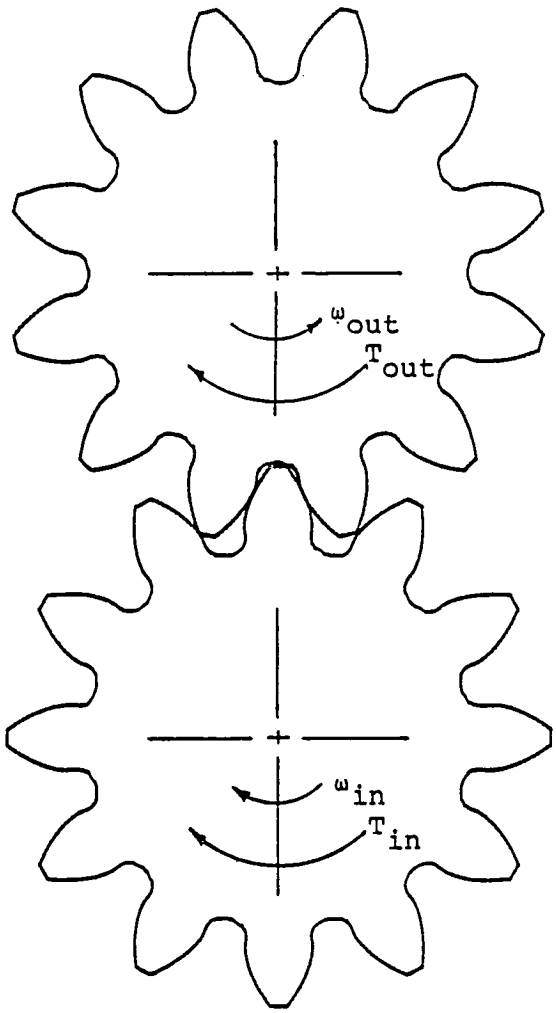


Figure 9 - Conventional Gear Train

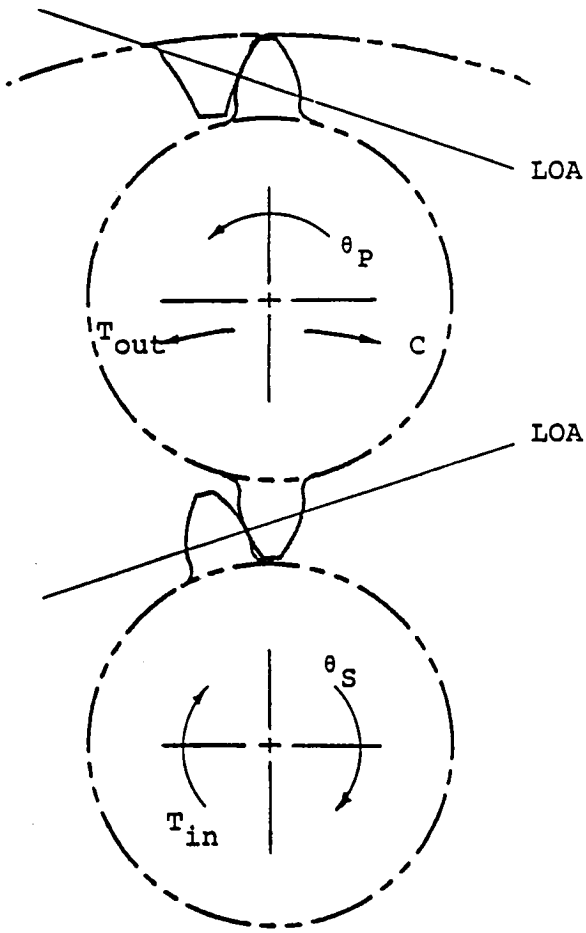


Figure 10 - Meshing Relationships in a PGT

Fv_e is not equal to the input power at the driving gear.

The PGT shown schematically in Figure 11 has a speed ratio:

$$m_p = \frac{\omega_D}{\omega_H} = \frac{d_C}{d_A} + 1 \tag{3.1}$$

where: d_C = Diameter ring gear

d_A = Diameter sun gear

The value of m_p is positive, implying the input and output shafts rotate in the same direction. To eliminate the effect of the rotation of the planet gears around the sun, an angular velocity, $-\omega_H$, is added to the entire system. The angular velocity of the planet carrier goes to 0, while the relative motions of all the gears in the train remain the same. The planet gears now become idlers and the entire assembly may be considered as a conventional gear train with fixed axes of rotation (Fig. 12).

No external forces or torques have been applied to the gear train. Therefore, the tangential forces acting on the gears of the fixed axes gear train, are identical to those in the actual PGT. Only the pitch-line velocities of the gears have been changed. With the pitch-line velocities being equal to the velocities of engagement in the PGT, the relative motion caused by carrier rotation has been eliminated, while maintaining the same transmitted forces between gears.

The absolute angular velocity of the input gear can be

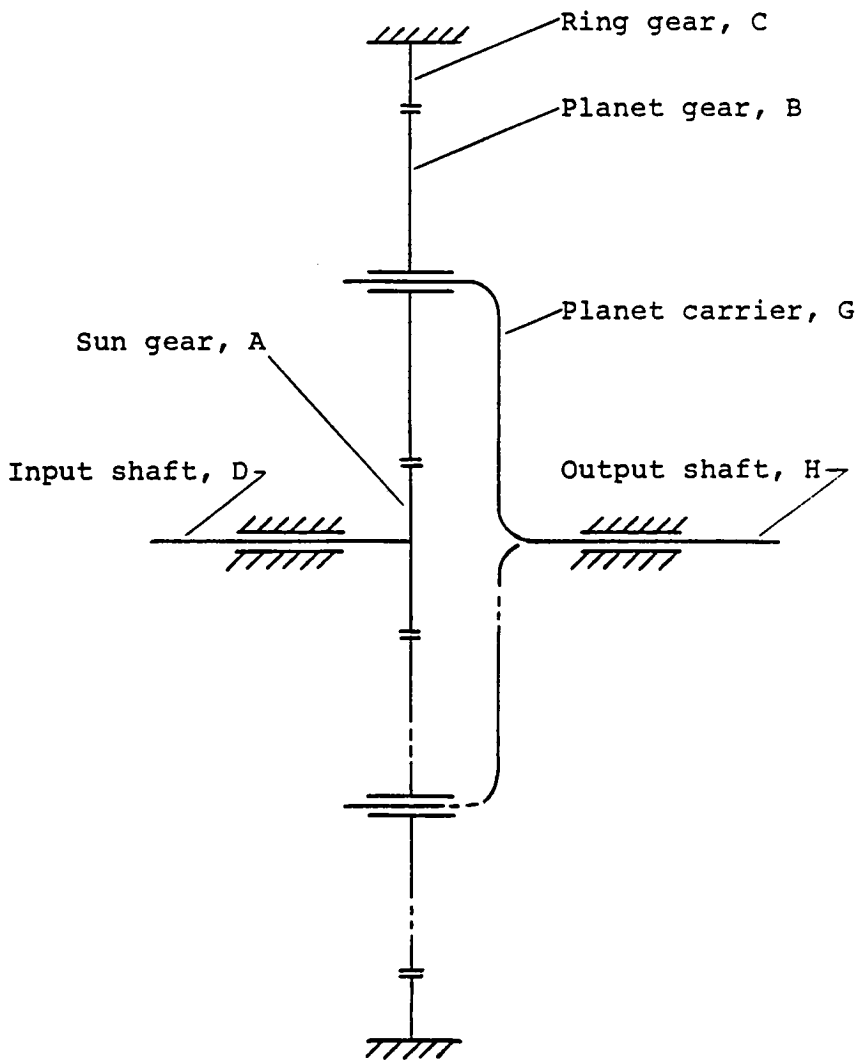


Figure 11 - Line Schematic of PGT

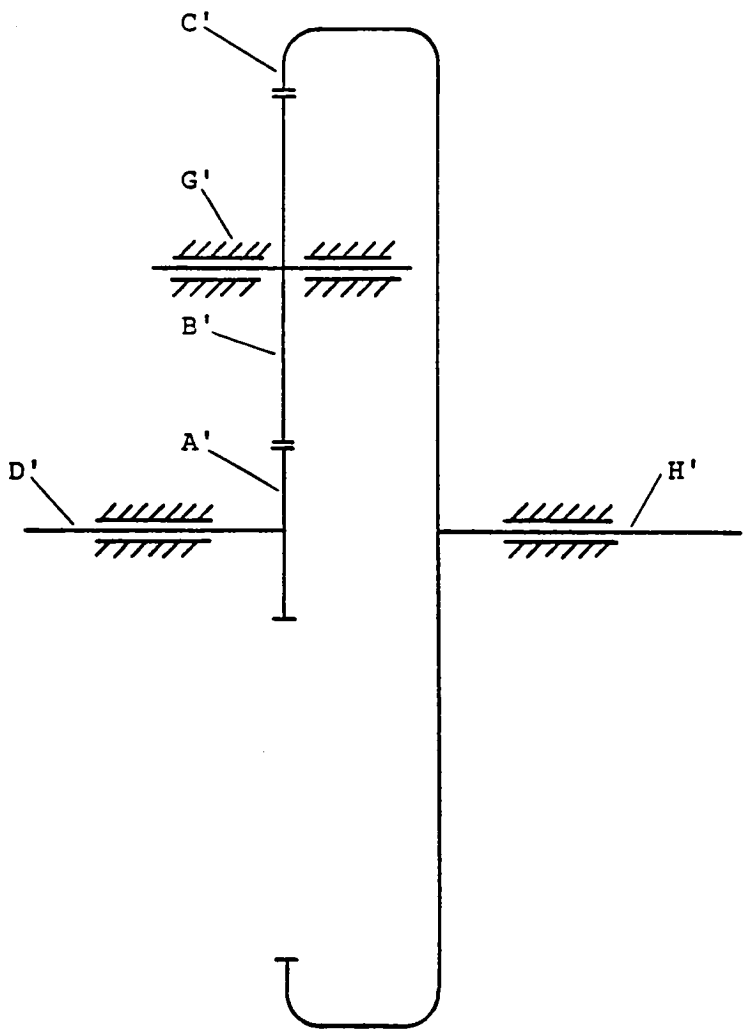


Figure 12 - Line Schematic of Conventional Gear Train

calculated as:

$$\omega_A' = \omega_A - \omega_H$$

$$= \omega_A \left(1 - \frac{1}{m_p}\right) \tag{3.2}$$

The angular velocity of the output shaft now becomes,

$$\omega_H' = \left(\frac{d_C}{d_B}\right)\left(-\frac{d_B}{d_A}\right)\omega_A'$$

$$= -\frac{d_C}{d_A}\omega_A' \tag{3.3}$$

with the minus sign denoting opposite rotations.

Note that since the ring gear is always larger than the sun gear, in Equation 3.1, the value of m_p will always be greater than one. Consequently, the velocities of engagement of gears in a PGT are less than those in a corresponding conventional gear train comprised of the same gears having the same input angular velocity and power. Since the tangential forces acting on the gears remain the same, the power developed by a PGT is lower than in a conventional gear drive. This implies that the dynamic tooth loads and wear, which are functions of the transmitted force and velocity of engagement, should be reduced in the PGT.

A kinematic examination of the PGT validates the equivalent engagement approach. This can be shown by selecting an arbitrary contact position along the line of action between the sun gear and a planet gear (Fig. 13). As the sun rotates through an angle θ_s, the rotation of the

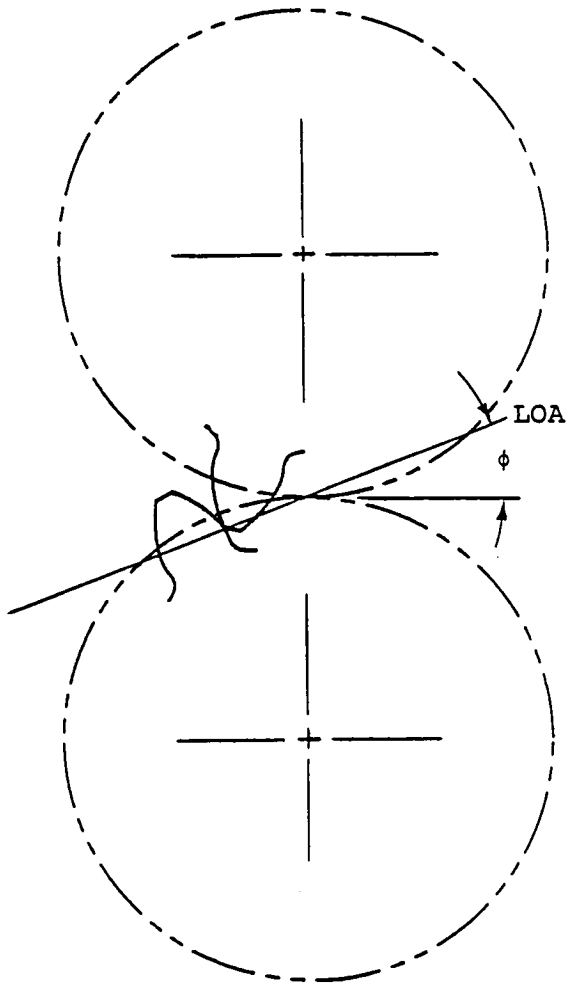


Figure 13 - Start of Contact

output carrier is:

$$\theta_c = \frac{N_S}{N_R + N_S}\,\theta_S. \tag{3.4}$$

Through the carrier rotation θ_c, equal arcs are swept out at the planet/ring contacting gear pair,

$$RPC_R\,\theta_C = -RPC_P\,\theta_P. \tag{3.5}$$

The minus sign is required since the planet and carrier rotate in opposite directions. The angular displacement of the planet gear about its own axes is:

$$\theta_P = -\frac{RPC_R}{RPC_P}\,\theta_c$$

$$= -\frac{N_R}{N_P}\,\theta_c$$

$$= -\frac{N_R N_S}{N_P(N_R + N_S)}\,\theta_S. \tag{3.6}$$

From Fig. 14, the sun/planet contacting gear pair have advanced along the line of action through an angle θ_S', due to the rotation θ_S. From equivalent arc lengths,

$$RBC_S\,\theta_S' = -RBC_P\theta_P. \tag{3.7}$$

Thus,

$$\theta_S' = -\frac{RBC_P}{RBC_S}\,\theta_P$$

$$= -\frac{N_P}{N_S}\left[-\frac{N_R N_S}{N_P(N_R + N_S)}\right]\theta_S$$

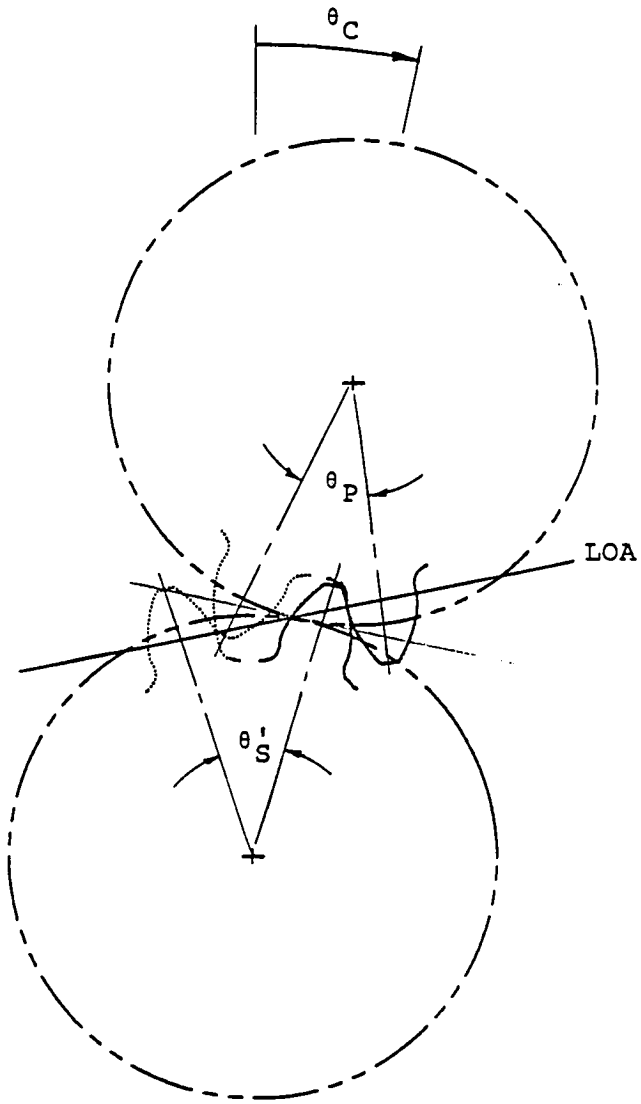


Figure 14 - Advancement of Contact Point
along the Line of Action

$$\theta_S' = \frac{N_R}{N_R + N_S}\,\theta_S$$

$$= \left(1 - \frac{1}{m_p}\right)\theta_S . \qquad (3.8)$$

Due to the carrier rotation, the contact point advancement is retarded along the line of action by the factor $(1 - \frac{1}{m_p})$, the same as derived in Eq. 3.1. Since no external forces have been applied, the analysis of the sun/planet and planet/ring static and dynamic behavior may be done by restricting the planet to rotate about its own axis, keeping the input torque the same, and calculating a new, effective input angular velocity at the sun gear.

Although this investigation is primarily concerned with planetary gear drives, thc concept of the equivalent conventional gear train can be extended to also include the star and solar gear arrangements. The same principles and techniques may then be used to evaluate those drives (Appendix B).

3.4.2.1 Phase Relationships Between Sun/Planet/Ring Mesh Stiffnesses

As stated in the literature survey, the main sources of dynamic excitation in geared systems are the variable-variable mesh stiffness (VVMS) functions and their discontinuities. In both parallel and co-axial shafted spur gear transmissons, the operational gear mesh stiffness is probably the single most important element in performing the static and dynamic analysis. Even in theoretically errorless

gearing, discontinuities in the mesh stiffness exist caused by the periodically changing number of teeth in contact. The only exception to this would be the case where the contact ratio is a whole number, implying that a gear tooth pair initiates contact simulaneously as the preceding tooth pair is concluding contact. In this manner, the number of teeth in contact would remain constant. Additional discontinuities are introduced when gear tooth errors exist which interrupt contact, thus altering the normal gear mesh stiffness characteristics. Increased system compliance and transmitted load also affect the mesh stiffness by increasing pre- and post-mature contact and decreasing the effect of gear tooth errors. An assumption that mesh stiffness values remain constant, or even that the mesh stiffnesses are equal for the sun/planet and planet/ring engagements, is completely unacceptable for realistic analyses of planetary gear trains.

Determination of the mesh stiffness as a function of fully loaded and deflected gear teeth in both the external-external and external-internal meshes (Fig. 15) must be completed before attempting any further analysis. The methodology used in determining the VVMS is described in Appendix A. To accurately determine the VVMS, the following parameters must be considered:

a. Magnitude of transmitted load
b. Gear tooth profile errors
c. Position of contact
d. Number of contacting tooth pairs

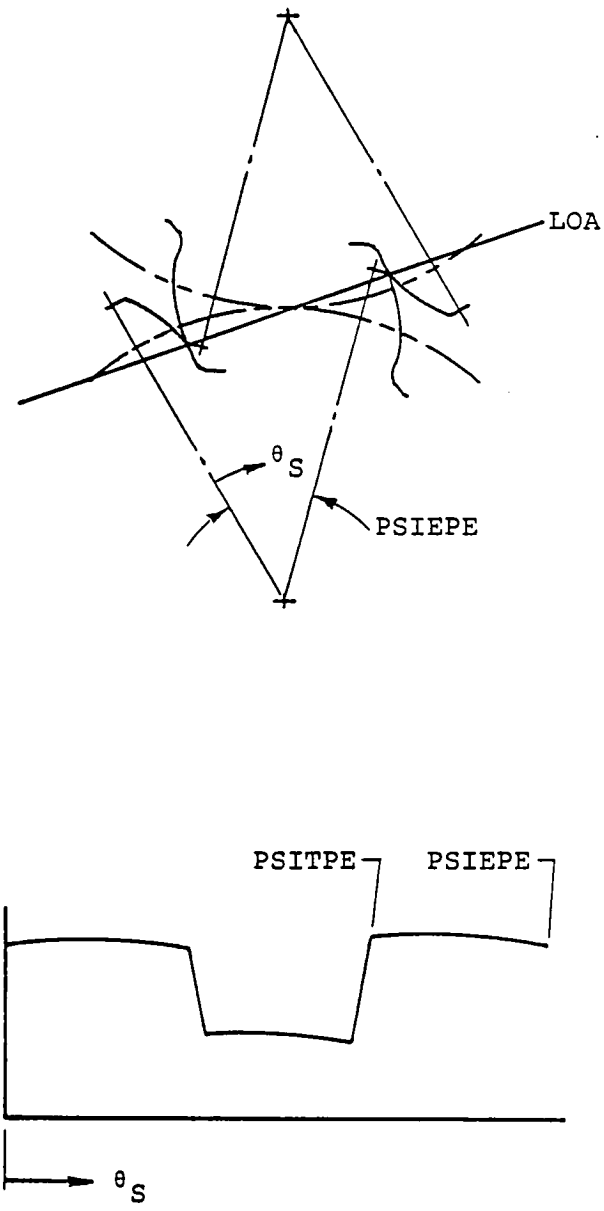


Figure 15 - Mesh Stiffness as a Function of Engaged Gear Tooth Position

e. Operational contact ratio

f. Gear and hub geometry

In Figure 7, three planet gears are shown symetrically situated about the sun gear. Although equally spaced, the planet gears generally do not share identical contact conditions with each other. This can be explained by examining the number of teeth on the sun and ring gear, respectively. If the number of teeth is not evenly divisible by the number of planets, the planets are not spaced a whole number of tooth pitches apart. Subsequently, different contact conditions must exist for each planet. The difference in contact includes the position of contact on the meshing teeth as well as the number of teeth in contact. In fact, it is often clearly desirable to have nonidentical conditions to minimize the variation of the mesh stiffness about the sun and planet gears [13,14].

This model can determine the magnitude and phase relationships of the mesh stiffnesses at all load transfer points. These relationships will affect both the static and dynamic behavior of the PGT. For example, in a low speed, quasi-static operation, the sun gear will move, if allowed, in a manner such that loading is equally distributed among the planet gears. The motion of the sun will be governed by the sun/planet mesh forces and the supporting mesh stiffnesses. Dynamically, the VVMS's phasing directly affects the excitation in the drive train, especially at the mesh frequency and its higher harmonics.

Phasing of the VVMS's depends on maintaining a static equilibrium condition about the sun gear such that the sun gear mesh forces collectively create a torque equivalent to the input torque. Also, the sum of the torques acting on each planet must be zero. This way, the planets experience no net torque, and the hub deformations on the planet at the sun/planet and planet/ring meshes remain localized. Planet gear teeth outside the two contact zones are thus, unaffected.

The phase relationships in the loaded mode between the individual gear tooth meshes are established by analyzing the planet positions about the sun gear. These are used to determine the i^{th} sun/planet and planet/ring mesh stiffnesses based on the instantaneous angular positions of the sun and planet, respectively. The first planet is positioned such that one incoming tooth pair is initiating contact between the sun and the planet. This position is then used as the reference point for the remaining planets.

The values of the fully loaded VVMS's for the i^{th} planet's engagement states are referenced against this position by:

$$K_{SPi}(\theta_S) = K_{SP1}(\theta_S + \phi_{Si}) \tag{3.9}$$

$$K_{PRi}(\theta_P) = K_{PR1}(\theta_P + \phi_{Pi} + \phi_{SR}), \tag{3.10}$$

where: θ_S = Instantaneous sun gear position.

K_{SPi} = Sun/planet loaded mesh stiffness as a function of θ_S.

ϕ_{Si} = Phase angle for the i^{th} sun/planet mesh.

θ_P = Instantaneous planet gear position.

K_{PRi} = Planet/ring loaded mesh stiffness as a function of θ_P.

ϕ_{Pi} = Phase angle for the i^{th} planet/ring mesh.

ϕ_{SR} = Phase angle between the start of contact at the sun/planet mesh and the start of contact at the planet/ring mesh.

The plase angles, ϕ_{Si} and ϕ_{Ri}, are based on the spacing of the planets about the sun gear (Fig.16) and the individual loaded gear mesh stiffness cycles. These are calculated by:

$$\phi_{Si} = \frac{\beta i}{2n} * PSITPE_i \qquad (3.11)$$

$$\phi_{Pi} = \frac{\beta i}{2n} * PSITPI_i \qquad (3.12)$$

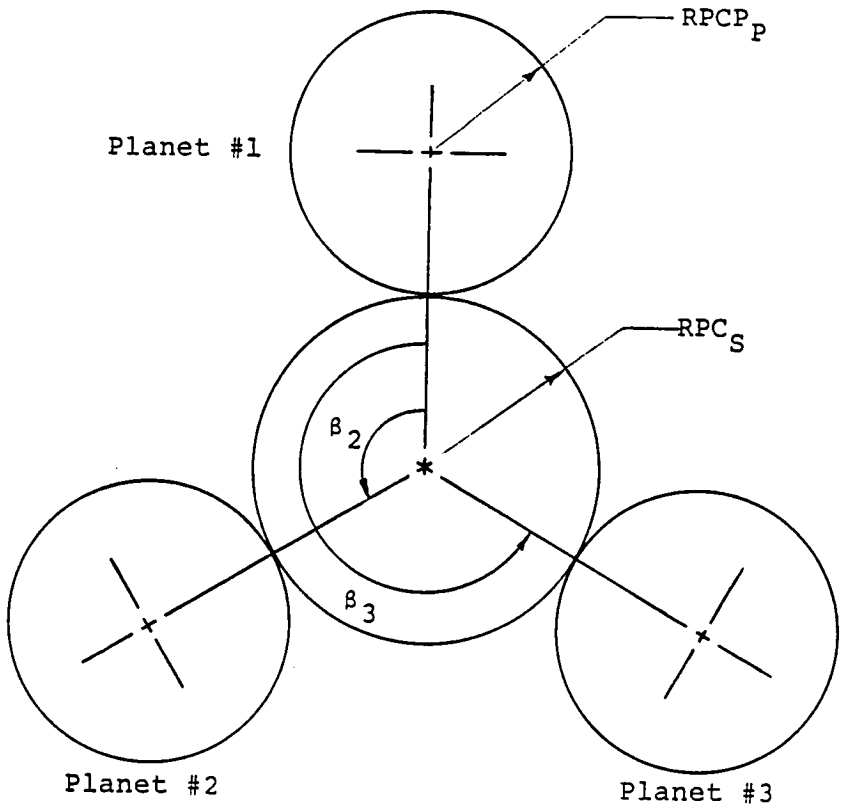
where: $\beta_i = \frac{2\pi(i-1)}{3}$

$PSITPE_i$ = Load sun/planet gear mesh stiffness cycle, radians.

$PSITPI_i$ = Loaded planet/ring gear mesh stiffness cycle, radians.

The phase angle, ϕ_{SR}, relates the engagement state at the reference planet between the sun/planet and planet/ring meshes. The angle is calculated through an iterative process which considers the theoretical, undeflected contacting gear teeth positions and then the actual deflected positions.

The first step requires positioning the reference planet such that an incoming tooth j would be initiating contact at



$$\beta_i = \frac{2\pi(i-1)}{3}$$

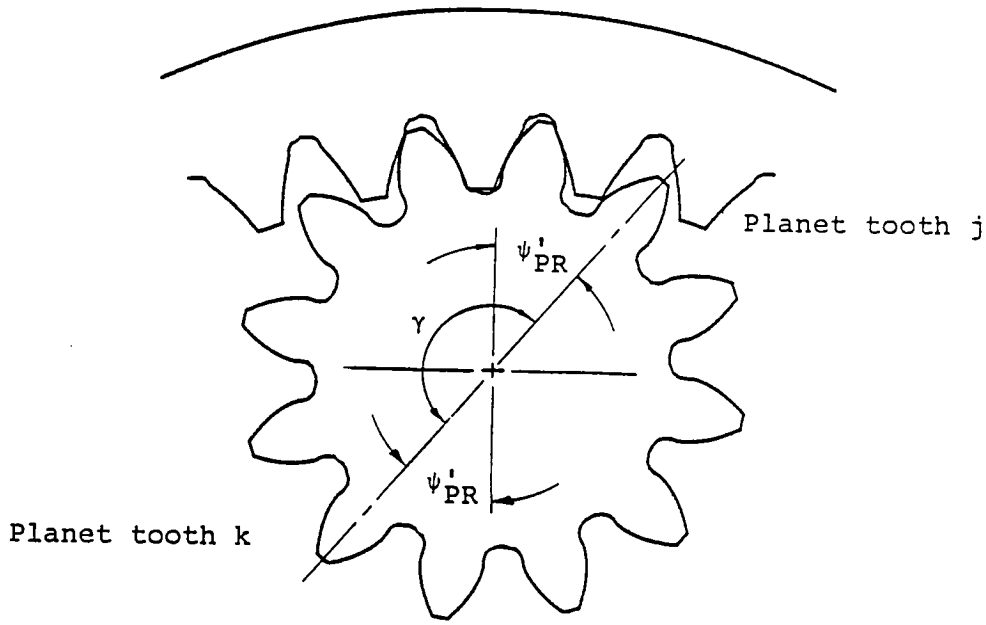
Figure 16 - Position of Planets about the Sun Gear

theoretical, no-load conditions (Fig. 17). The deflected and undeflected centerline angles of the initiation of gear tooth contact at the sun/planet and planet/ring engagements have been previously determined absolutely as detailed in Appendix A. Since this orientation is done under no-load conditions, the position of tooth k is also determined. Next, the position of tooth j is adjusted to reflect the actual, load conditions (Fig. 18). Since the deflections are assumed to be localized, the position of tooth k remains fixed. Finally, the planet is rotated until tooth k has reached the position of initiating fully loaded contact (Fig. 19). Concurrently, tooth j has advanced by an amount of ϕ_{SR} radians, which now becomes the phase angle relating the contact conditions between the sun/planet and planet/ring meshes.

Knowing the phase relationships of all the individual meshes allows the static loading and positioning to be determined simply as a function of the angular position of the sun gear. This is an important index as it will allow the monitoring of the rotation of seven elements and the two-dimensional translation of the sun gear through one independent variable. The techniques used in the static analysis are implemented in the investigation of the dynamic behavior of the PGT.

3.4.2.2 Incorporating Planet Positioning Errors

Any attempted modeling of a physical system must acknowledge the existence of errors and allow for them in

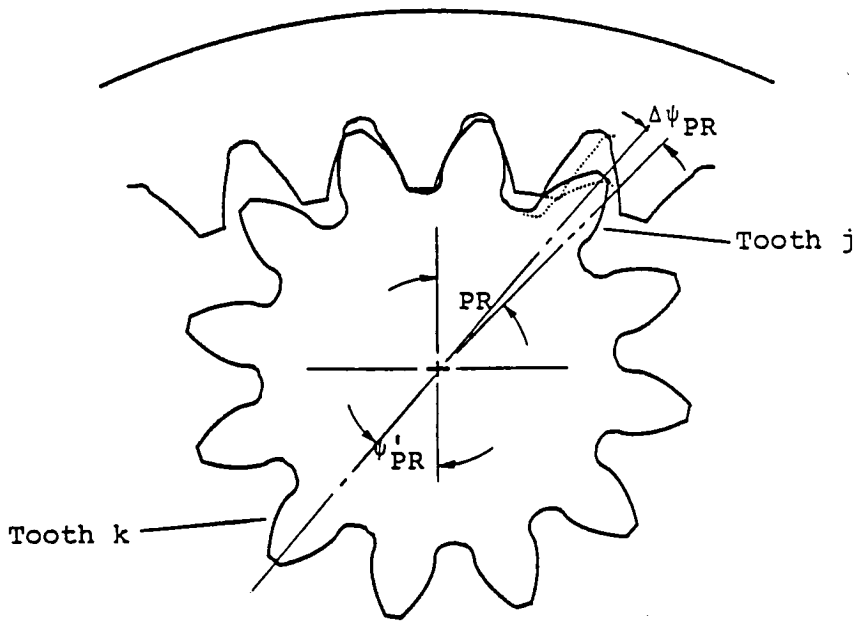


ψ'_{PR} = Angle at start of undeflected contact

γ = π, if TGP is even

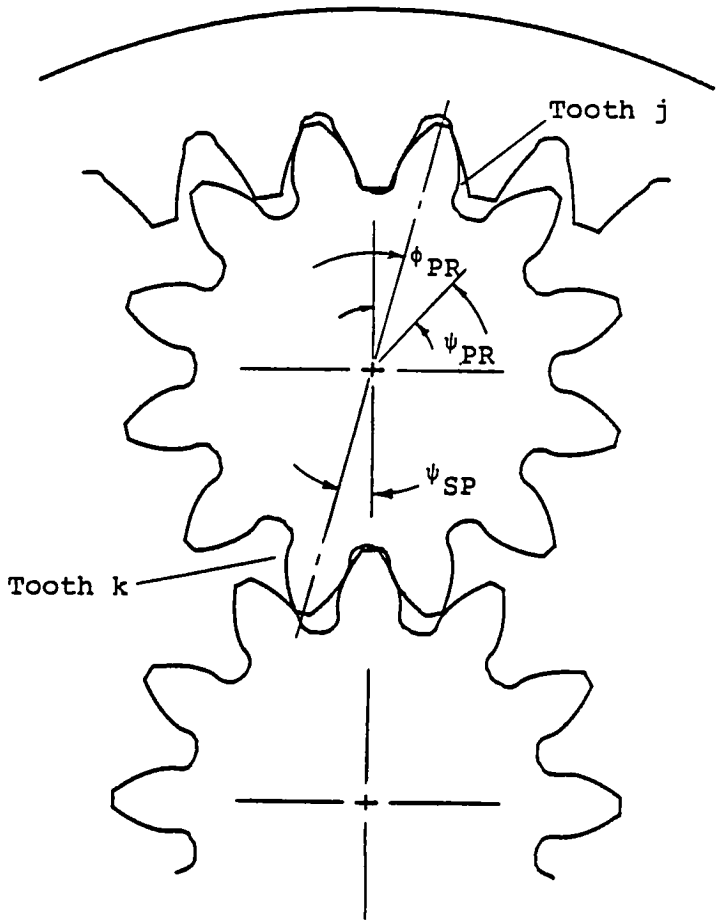
γ = $\pi - \frac{TGP}{4\pi}$, if TGP is odd

Figure 17 - Tooth j Initiates No-Load Contact



ψ_{PR} = Angle at start of deflected contact

Figure 18 - Tooth j Initiates Fully-Loaded Contact



ψ_{SP} = Angle at start of deflected contact

Figure 19 - Tooth k Initiates Fully-Loaded Contact

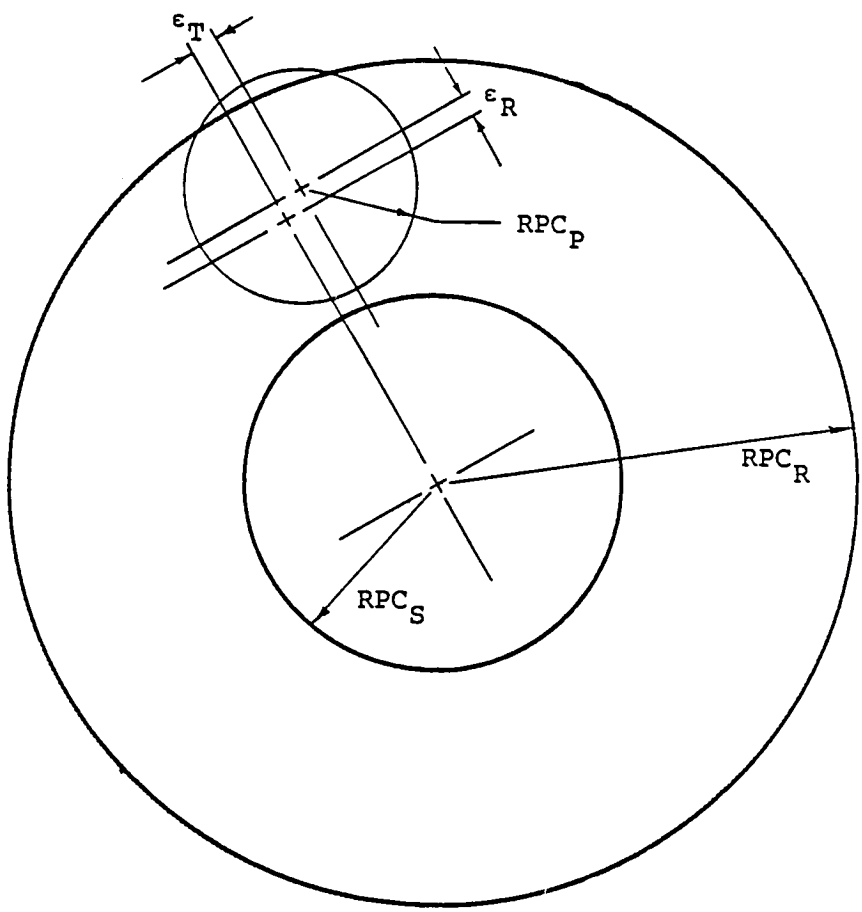
their simulation or offer sufficient evidence that these errors are negligible, their effect is small in comparison with other disturbances, and therefore, may be omitted. Significant system errors which may occur through the quality of manufacturing and assembly of PGT's are:

a. Gear geometry errors
b. Radial positioning errors of planets.
c. Tangential positioning errors of planets

Since manufacturing techniques are inexact, gears do not possess the ideal involute geometry. Instead, they may have, to varying degrees, deviations from the true involute, thickness error, spacing error, or even develop errors such as pitting after prolonged use. These errors may interrupt contact, introducing additional dynamic excitations in the system. In this model, it is not necessary that all the planets reflect the same gear error. By separately establishing the mesh stiffness characteristics for each planet, it is possible to examine the effect on gear errors on just one planet, rather than identical errors on all n planets.

Figure 20 shows a planet with tangential, ε_T, and radial, ε_R, positioning errors. These errors are measured from the position the planet would occupy if it were properly located. The magnitude of the combined effect of these errors can be taken as the eccentricity, ε, where:

$$\varepsilon = \sqrt{\varepsilon_T^2 + \varepsilon_R^2} \qquad (3.13)$$



RPC_P - Pitch circle radius, planet
RPC_S - Pitch circle radius, sun
RPC_R - Pitch circle radius, ring

Figure 20 - Tangential and Radial Positioning Errors

Previous investigators have generally ignored eccentricity errors since tooth errors are found to have a much greater effect on dynamic loading [11, 21]. This investigation will consider the tangential component since this directly affects the phasing of the mesh stiffnesses among the components. It can be easily shown that the effects of the radial component of the positioning error is indeed small in comparison with tooth errors.

Investigations by Richardson, Kasuba, and Harris, have shown that for high quality gears, errors present are sinusoidal in nature. Thus:

$$\varepsilon_t = \varepsilon_{tmax}\sin(\omega t - \alpha) \tag{3.14}$$

$$\varepsilon_r = \varepsilon_R\sin(\dot{\theta}_p t - \gamma) \tag{3.15}$$

where: ε_t = tooth error

ε_r = radial error

ω = tooth engagement velocity

$\dot{\theta}_p$ = angular velocity of gear

α,γ = phase angles of errors with respect to correct tooth position

The frequency of tooth error is the same as the tooth engagement frequency and can be found from the number of teeth on the planet (TGP) and the angular velocity by:

$$\omega = TGP * \dot{\theta}_p. \tag{3.16}$$

The frequency of radial error is the same as speed of rotation since there is one cycle of radial error for every

rotation of the planet about its axis. The resulting accelerations caused by tooth and radial errors can be found by differentiating Eqs 3.13 and 3.14 twice with respect to time:

$$\frac{d^2\varepsilon_t}{dt^2} = -\varepsilon_{tmax}\omega^2 \sin(\omega t - \alpha) \tag{3.17}$$

$$= -\varepsilon_{tmax} TGP^2 \theta_p^2 \sin(TGP * \dot{\theta}_p - \alpha)$$

$$\frac{d^2\varepsilon_r}{dt^2} = -\varepsilon_R \theta_P^2 \sin(\dot{\theta}_p - \gamma) \tag{3.18}$$

If TGP = 28, and $\varepsilon_{tmax} = 0.1_R$, the accelerations caused by the tooth errors are still almost two orders of magnitude larger than those caused by radial error. Consequently, radial errors may be ignored [26].

While the effect of the radial error has been dismissed by previous studies [11], the tangential error may not be so lightly dismissed. The reason for this is the interdependence of all the sun/planet/ring mesh stiffnesses. The tangential errors could cause either a lead of a lag error in the engagement characteristics of a particular planet. Also, some commercial designs, such as the Stahl-Laval model CPG, purposely use asymmetrically spaced planets. The tangential error can be incorporated into the model by readjusting the planet spacing calculation in Eqs. 3.11 and 3.12 (Fig. 21). If,

$$\theta_{\varepsilon i} = \frac{\varepsilon t}{RPC_S + RPC_P} \tag{3.19}$$

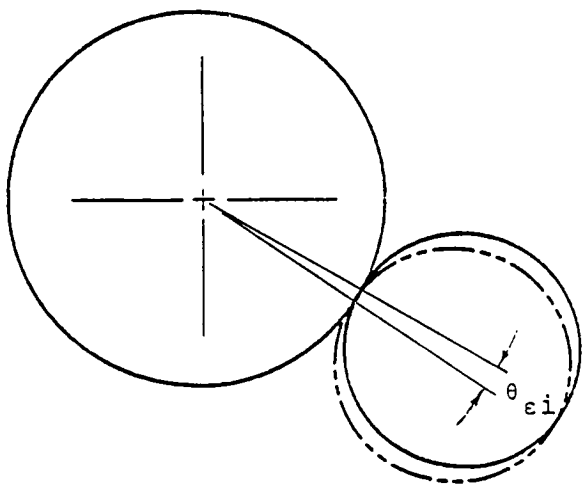


Figure 21 - Modeling Tangential Positioning Error

Then β_i the spacing angle, becomes;

$$\beta_i = \frac{2\pi(i-1)}{3} + \theta_{\varepsilon_i} \tag{3.20}$$

The sign on $\theta_{\varepsilon i}$ is positive if the planet leads its design position, negative if it lags.

3.4.3 Synchronous Positioning of PGT's

An important parameter in the modeling of a system is the determination of the time period in which physical events re-occur. Since this investigation is concerned with the steady-state operation of a PGT, it is necessary to find the period, τ, such that:

$$S(\theta_S + \tau\dot{\theta})_S = S(\theta_S), \tag{3.21}$$

where S is a function of system response based on loading and displacement, and θ_S is the input predicated by the monitored position of the sun gear. Thorough examination of the system through one period would then be sufficient to develop a time history for the system, and to make time based judgements on design parameters such as fatigue failures.

A static evaluation through one period must be done to establish baseline values for the mesh forces against which the dynamic forces will be compared. The time period is important dynamically for two reasons. One, it determines how long the differential equations describing the system must be evaluated to eliminate transient behavior and assure steady state operation. Second, it determines the time increment required for stability during the numerical

integration of those equations.

A synchronous position is defined as one where a predetermined state of engagement between all 2n gear meshes is repeated, and the period is defined as the elapsed time between synchronous positions. In the case of the PGT shown in Figure 7, the synchronous position would occur when planet #1 returns to the 12 o'clock position, with the #1 sun tooth engaged with the #15 planet tooth, and the Nos. 2 and 1 planet teeth engaged with the Nos. 1 and 70 ring teeth. For the particular case shown, for one revolution of the output shaft, the planet rotates 2.5 times about its own axis. Therefore, the output shaft must rotate twice and the planet five times about its own axis to duplicate the initial engagement state. The five planet rotations mean there are five times 28, or 140 engagement cycles between synchronous positions. The engagement cycle is defined as the mesh cycle which is extended due to pre- and post-mature contact caused by elastic deflections.

For the case where the planets have identical errors, and therefore, identical stiffness functions, the period of the synchronous position is shortened to just the period of the meshing cycle. Since there are no singular planet errors, the synchronous position is repeated with the beginning of contact of two gear teeth between the sun and reference planet.

3.4.4 Static Load Sharing Among Planet Gears

Previous experimental investigations have confirmed the

fact that allowing the sun gear to float tends to equalize the load distribution among the planet gears. It is the purpose of this section to formally develop a static model based on a sun gear with variable fixity, mathematically show that for static conditions this model correctly predicts the proper load sharing, and further refine the model so it can be used in an efficient, computer algorithm.

Other studies [19, 24] have assumed that the sun gear was not centered and determined equivalent torsional models. The sun motion was then calculated based on influence coefficients from all the mesh stiffness. This technique affords no way of allowing for adjustments in the sun support. Also it cannot be used to determine planet load sharing if the support is soft, i.e., the support is not stiff enough to fix the sun center, but it is stiff enough that it is not negligible.

Figure 22 shows the static model of the equivalent conventional gear train used for improving this study. The sun and ring mesh stiffnesses are represented by k_{si} and k_{ri}, respectively. As described in the preceding section, these values are functions of θ_s, the angular position of the sun gear. The angular position will be some $j^{\underline{th}}$ interval of one synchronous cycle. The sun support stiffness is modeled by two equal perpendicular springs, k', whose values can be varied to simulate different degrees of fixity on the sun's center. Figure 23 shows the coordinate system used in the static development.

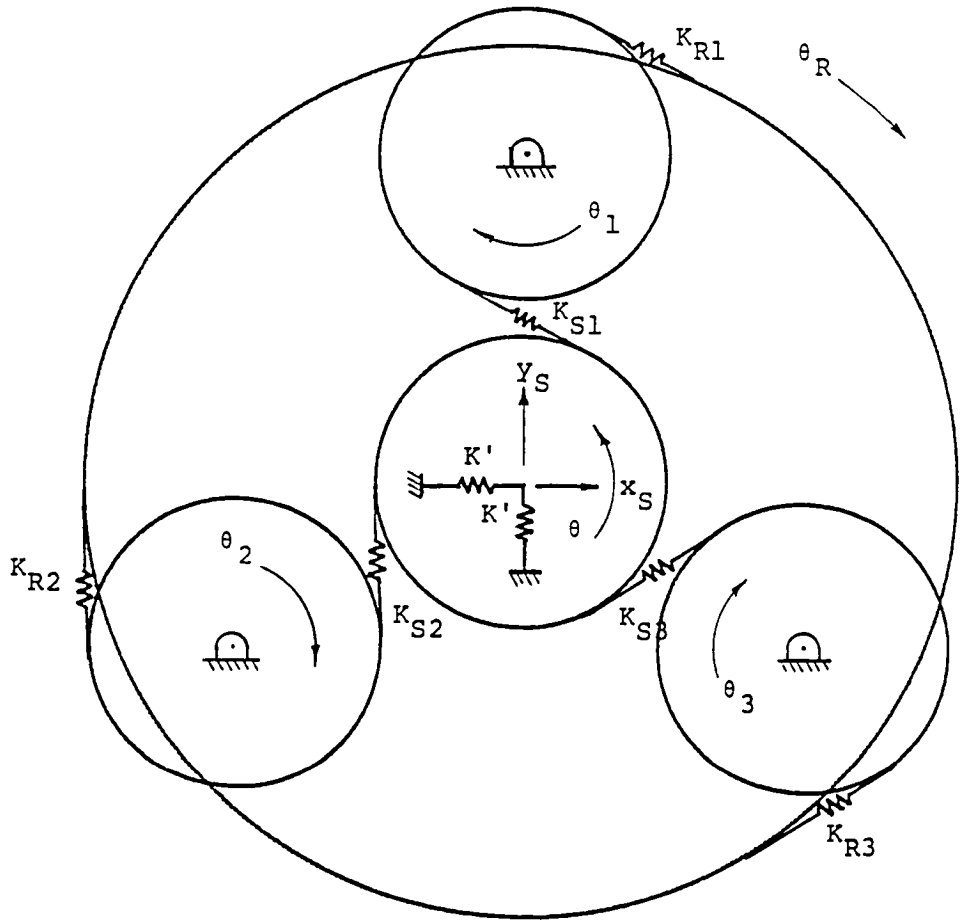


Figure 22 - Static Model of Equivalent PGT

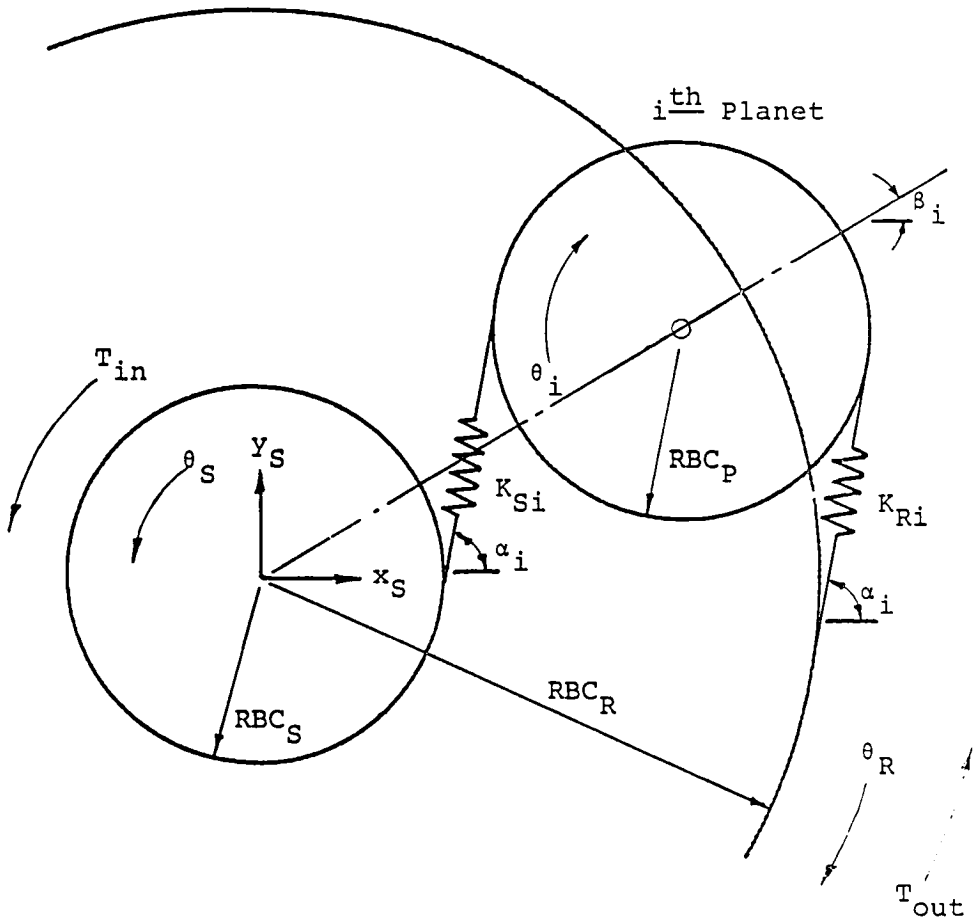


Figure 23 - Coordinate System

The static analysis determines the motion of the components relative to the sun gear. Therefore, θ_S may be taken as zero. Only the torques about the planets and ring gear need to be considered to find their relative angular displacements and the translation of the sun gear. Summing up the torques about the ring when the sun is at the $j^{\underline{th}}$ position gives:

$$\sum_{i=1}^{n} RPC_R[K_{Rij}(RPC_P\theta_{ij} - RPC_R\theta_{Rj})] = T_{out} \qquad (3.22)$$

The forces acting on the $i^{\underline{th}}$ planet are affected not only by the relative angular displacements, but also by the translation of the sun. Summing up the torques about the $i^{\underline{th}}$ planet gives:

$$RPC_PK_{Rij}(RPC_R\theta_{Rj} - RPC_P\theta_{ij}) + RPC_PK_{Sij}(RPC_S\theta_{Sj} - RPC_P\theta_{ij}) + RPC_PK_{Sij}x_{Sj}\cos_i + RPC_PK_{Sij}Y_{Sj}\sin\alpha_i = 0 \qquad (3.23)$$

Finally considering the static equilibrium of forces on the sun gear in the x and y directions gives:

$$-K_Sx_{Sj} + K_{Sij}[(RPC_P\theta_{ij} - RPC_S\theta_S) - x_S\cos\alpha_i - y_S\sin\alpha_i]\cos\alpha_i = 0 \qquad (3.24)$$

$$-K_Sx_{yj} + K_{Sij}[(RPC_P\theta_{ij} - RPC_S\theta_S) - x_S\cos\alpha_i - y_S\sin\alpha_i]\sin\alpha_i = 0 \qquad (3.25)$$

These equations can be combined in matrix form:

$$[K]\{\Phi\}_j = \{V\} \qquad (3.26)$$

where: Φ = displacement vector

$$= [\theta_r, \theta_1, \ldots, \theta_n, x_S, y_S]^T$$

V = force vector

$$= [-T_{out}, 0, \ldots, 0, 0, 0]^T$$

K = stiffness matrix (Fig. 24)

The relative displacement vector, Φ, can now be solved simply by using a Gaussian elimination and back substitution technique. The matrix symmetry also allows the stiffness matrix to be quickly assembled.

Since the sun rotation is 0, and the mesh force is calculated by the relative displacement along the line of action, the i^{th} planet load for the j^{th} position of the sun gear is:

$$F_{ij} = K_{Sij} REC_P \theta_i \tag{3.27}$$

With the planet being in static equilibrium, this force must be equivalent to the force transmitted by the planet/ring mesh. Finally, to check whether the proper relative motions and transmitted forces have been calculated, the input and output torques must be balanced through the following relationships:

$$T_{in} = \sum_{i=1}^{n} F_{ij} * RBC_S \tag{3.28}$$

$$T_{out} = \sum_{i=1}^{n} F_{ij} * RBC_R \tag{3.29}$$

The sun gear bearing support reactions are:

$$
K = \begin{bmatrix}
RPC_R^2 \Sigma K_{Ri} & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
-RPC_R RPC_P K_{Ri} & RPC_P^2 (K_{S1} + K_{R1}) & \cdot & \cdot & \cdot & \cdot & \cdot \\
-RPC_R RPC_P K_{R2} & 0 & RPC_P^2 (K_{S2} + K_{R2}) & \cdot & \cdot & \cdot & \cdot \\
\vdots & \vdots & 0 & \ddots & \cdot & \cdot & \cdot \\
-RPC_R RPC_P K_{Rn} & 0 & 0 & \cdots & RPC_P^2 (K_{Sn} + K_{Rn}) & \cdot & \cdot \\
0 & -RPC_P K_{S1} \cos\alpha_1 & -RPC_P K_{S2} \cos\alpha_2 & \cdots & -RPC_P K_{Sn} \cos\alpha_n & K' + \Sigma K_{Si} \cos^2\alpha_i & \cdot \\
0 & -RPC_P K_{S1} \sin\alpha_1 & -RPC_P K_{S2} \sin\alpha_2 & \cdots & -RPC_P K_{Sn} \sin\alpha_n & \Sigma K_{Si} \cos\alpha_i \sin\alpha_i & K' + \Sigma K_{Si} \sin^2\alpha_i
\end{bmatrix}
$$

(matrix is symmetric)

Figure 24 - Stiffness Matrix

$$F_{xj} = \sum_{i=1}^{n} F_{ij}\cos\alpha_i \tag{3.30}$$

$$F_{yi} = \sum_{i=1}^{n} F_{ij}\sin\alpha_i \tag{3.31}$$

3.5 Dynamic Analysis

The gear train to be used for the dynamic analysis is shown in Figure 25. The gear train is comprised of a power source, load, the equivalent planetary transmission, and the input and output shafts and their bearings. It is capable of considering fluctuating output torque: damping in shafts, gears and bearings; non-involute gear meshes; and loss of contact between gear teeth. The dynamic model of this gear train has a total of nine degrees of freedom. The sun gear has three degrees of freedom, two transverse and one rotational. The driver, planets, ring, and the load have one degree of freedom (rotation). It was assumed that this was the minimum number of allowable motions to accurately simulate the gear train behavior without making the analysis overly cumbersome. Assumptions for restrictions on the motion of the planet gears seem valid since the bending stiffnesses of the planet shafts are generally much greater than the mesh stiffnesses. According to experimental results [20], the displacement of a ring gear as a rigid body is small in the radial direction. So, it can be assumed that the ring gear has no planar motion. The coordinate system given in Figure 23 is also used in the dynamic analysis.

1. Driver, J_D
2. Input shaft
3. Sun gear, J_S
4. Planet gear, J_P
5. Ring gear, J_R
6. Carrier, J_C
7. Output shaft
8. Load, J_L

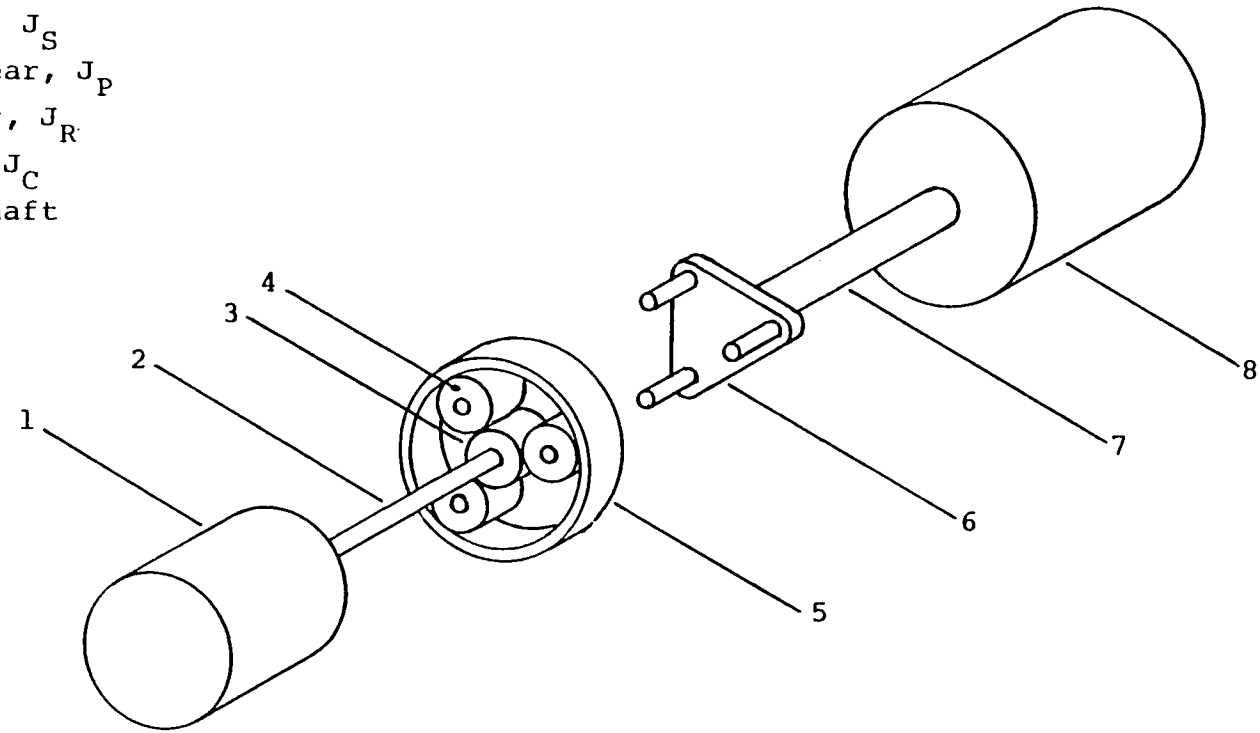


Figure 25 - Gear Train Model Used for Conventional Gear Train

The equations of motion used to calculate the instantaneous dynamic loads are based on the equivalent companion gear train shown in Figure 15. By adding the negative angular velocity of the carrier to the angular velocities of the components shown in Figure 25, the effect of the carrier rotation on the pitch-line velocities is eliminated. Therefore, the equations of motion need only to consider the effective engagement velocities.

The inertial influence of the carrier on the dynamic loads can be preserved by assigning an equivalent moment of inertia to the ring gear, which now becomes the transmission output element. The equivalent ring gear inertia should be a function of the planet and carrier masses.

The equations of motion developed for the companion gear train use the instantaneous parameters determined during the static analysis for various mesh arc positions. These parameters are based on the instantaneous rather than the theoretical line of action. Dynamically, their effect is to change the transmission ratio to reflect the actual contact condition as influenced by deformation in the contact zone and tooth profile errors. For example, if contact occurs above the intended line of action, the effective base circle radius of the driven gear is reduced, decreasing the output torque. This can be expressed as:

$$RBC' = RBC * \eta_{P/G} \tag{3.32}$$

$$T_{out}' = T_{out} * \eta_{P/G'} \tag{3.33}$$

where: RBC' = instantaneous base circle radius

RBC = theoretical base circle radius

T_{out}' = instantaneous output torque

T_{out} = theoretical output torque

$\eta_{P/G}$ = per cent effective reduction due to pinion/gear deformation

For the PGT modeled, the input torque will be assumed to be constant, while the output torque will vary according to the change in the effective base circle radii of the planet and ring gears, respectively. The derivation of the instantaneous output torque can be started by examining the equivalent conventional gear train with just one planet. If all components were considered rigid, the output torque would be:

$$T_{out} = \frac{RBC_R}{RBC_P} * \frac{RBC_P}{RBC_S} * T_{in} \qquad (3.34)$$

If the instantaneous transmission parameters due to profile errors and modifications, deflections, height of engagement, and angular position of engagement are now considered, the effective output torque becomes:

$$T_{out}' = \frac{RBC_R * \eta_{P/R}}{RBC_P} * \frac{RBC_P * \eta_{P/S}}{RBC_S} * T_{in}$$

$$= RBC_R * \eta_{P/R} * \eta_{S/P} * T_{in}/RBC_S, \qquad (3.35)$$

where: $\eta_{P/R}$ = per cent change of RBC_R due to planet/ring deformation

$\eta_{S/P}$ = percent change of RBC_P due to sun/planet deformation

The term T_{in}/RBC_S is the force transmitted along the line of action of the sun/planet mesh. Therefore, for an arrangement with n planets, the instantaneous output torque is:

$$T_{out}' = RBC_R * (\eta_{P/R} * \eta_{S/P} * GPTFE)_j \qquad (3.36)$$

where: GPTFE = sun/planet gear pair transmitted force based on load sharing

j = instantaneous values at the j^{th} position of the sun gear

The differential equations of motion for the components of the equivalent companion gear train in Figure 25 are:

Driver

$$J_D\ddot{\psi}_D + C_{BD}\dot{\psi}_D + C_{DS}(\dot{\psi}_D - \dot{\psi}_S) + K_{DS}(\dot{\psi}_D - \dot{\psi}_S) = T_{in} \qquad (3.37)$$

Sun Gear

$$J_S\ddot{\psi}_S + C_{BS}\dot{\psi}_S + C_{DS}(\dot{\psi}_S - \dot{\psi}_D) + K_{DS}(\psi_S - \psi_D) + \sum_{i=1}^{n} (RBC_S * GPTFE_i) = 0 \qquad (3.38)$$

$$M_S\ddot{x}_S + C_S\dot{x}_S + K_S x_S + \sum_{i=1}^{n} [GPTFE_i * \cos(\pi - \phi - \frac{(i-1)*2\pi}{n})] = 0 \qquad (3.39)$$

$$M_S\ddot{y}_S + C_S\dot{y}_S + K_S y_S + \sum_{i=1}^{n} [GPTFE_i * \sin(\pi - \phi - \frac{(i-1)*2\pi}{n})] = 0 \qquad (3.40)$$

Planet i

$$J_P\ddot{\psi}_i + C_P\dot{\psi}_i - RBC_P'(GPTFI_i - GPTFE_i) = 0 \qquad (3.41)$$

Ring

$$J_R\ddot{\psi}_R + C_{BR}\dot{\psi}_R + C_{LS}(\dot{\psi}_R - \dot{\psi}_L) + K_{LS}(\psi_R - \psi_L)RBC_R' + \sum_{i=1}^{n} (RBC_R' * GPTFI_i) = 0 \tag{3.42a}$$

$$J_R = (J_C + 3J_P) + (M_C + 3M_P)RPC_R^2 \tag{3.42b}$$

Load

$$J_L\ddot{\psi}_L + C_{BL}\dot{\psi}_L + C_{LS}(\dot{\psi}_L - \dot{\psi}_R) + K_{LS}(\psi_L - \psi_R) = -T_{out}' \tag{3.43}$$

The variables $GPTFE_i$ and $GPTFI_i$ are the gear pair transmitted force for the $i^{\underline{th}}$ external-external (sun/planet) and internal-external (planet/ring) meshes, respectively. They represent the dynamic response of PGT meshes to the excitation provided by the discontinuities in the mesh stiffnesses and changes in the transmission ratio. At each mesh, three situations can occur based on the relative dynamic motions of the involved gears. These are:

i) If $$RBC_j'\psi_j < RBC_k\psi_k$$

where: j = driven gear

k = driving gear

then normal operating situation exists, and the dynamic transmitted force is:

$$GPTFE_i = C_{SPi}[RBC_S*\dot{\psi}_S - RBC_P'*\dot{\psi}_i + x_S\cos\alpha_i + y_S\sin\alpha_i] + K_{SPi}[RBC_S*\psi_S - RBC_P'*\psi_i + x_S\cos\alpha_i + y_S\sin\alpha_i] \tag{3.44}$$

$$GPTFI_i = C_{PRi}[RBC_R*\dot{\psi}_R - RBC'_P*\psi_i] + K_{PRi}[RBC_R*\psi_R - RBC'_P*\psi_i] \quad (3.45)$$

$$\alpha_i = \pi - \phi - \frac{(i-1)*2\pi}{n}$$

ii) If $RBC'_j\psi_j > RBC_i\psi_i$

but, $(RBC'_j\psi_j - RBC_i\psi_i) < BACKLASH$

the gears have separated, disrupting contact between the gear teeth. The gears are now rotating independently of each other.

So,

$$GPTFE_i = 0 \quad (3.46)$$

$$GPTFI_i = 0 \quad (3.47)$$

iii) If $RBC'_j\psi_j > RBC_i\psi_i$

and, $RBC'_j\psi_j - RBC_i\psi_i > BACKLASH$

The driven gear has impacted on the backside of the driving gear tooth immediately preceding it, and the line of action is momentarily reversed. Therefore,

$$GPTFE_i = C_{SPi}*[RBC_S*\dot{\psi}_S - RBC'_P*\dot{\psi}_i + x_S\cos\alpha_i + y_S\sin\alpha_i] + K_{SPi}*[RBC_S*\psi_S - RBC'_P*\psi_i + x\cos\alpha_i + y\sin\alpha - BACKLASH] \quad (3.48)$$

$$GPTFI = C_{PRi}*[RBC_R*\dot{\psi}_R - RBC'_P*\dot{\psi}_i] * K_{PRi}*[RBC_R*\psi_R - RBC'_P*\psi_i - BACKLASH] \quad (3.49)$$

In cases i and iii, if the instantaneous K_{SPi} or $K_{PRi} = 0$, then the respective transmitted force is zero.

The equations of motion contain damping terms for the components in addition to the stiffnesses. The damping component in the bearings has been combined with the element located nearest that particular bearing, i.e., damping from the inboard drive bearing, C_{BS}, is included in the sun gear equation. Based on published experimental results [26], a critical damping ratio of $\zeta_S = 0.005$ has been used to calculate the effective shaft damping. The shaft masses have been taken to be the effective mass between the connected masses. Thus,

$$C_{DS} = 2\zeta_S \sqrt{\frac{K_{DS}}{\frac{1}{M_D} + \frac{1}{M_S}}} \tag{3.50}$$

$$C_{LS} = 2\zeta_S \sqrt{\frac{K_{LS}}{\frac{1}{M_R} + \frac{1}{M_L}}} \tag{3.51}$$

The critical damping ratio for engaged gear teeth, ζ_G, has been measured to range between 0.03 and 0.10 [6], [25]. The effective damping of the gear meshes is,

$$C_{SPi} = 2\zeta_G \sqrt{\frac{K_{SRi}}{\frac{1}{M_S} + \frac{1}{M_P}}} \tag{3.52}$$

$$C_{PRi} = 2\zeta_G \sqrt{\frac{K_{PRi}}{\frac{1}{M_P} + \frac{1}{M_R}}} \tag{3.53}$$

Since, the damping is shown as being a function of gear mesh, this term will change throughout the various mesh arcs.

The equations of motion (Eqns. 3.36 - 3.43) represent a system of nine, simultaneous, linear second-order differential equations with periodically varying coefficients. The initial displacements of the gears are determined using the technique described in Section 3.4.4. These values are found by solving for the vector Φ . The stiffness matrix, K, would be based on positioning the gears such that an arbitrary sun gear pair is initiating contact with the No. 1 planet.

The initial positions of the driver and load are referenced against the initial position of the sun gear, which was taken to be zero. Since the driving torque and driver displacement act in the same direction, the initial displacement of the driver is:

$$\psi_D = T_{in}/K_{DS} \tag{3.54}$$

The load torque acts in the opposite direction of the load displacement. Therefore, the initial displacement of the load is:

$$\psi_L = \psi_R - T_{out}/K_{LS} \tag{3.55}$$

where ψ_R is the initial absolute displacement of the ring gear. The initial angular velocities are set to be the nominal steady-state equivalent engagement velocities as determined in section 3.4.1. The initial translational velocities of the sun center are taken as zero.

The equations of motion are numerically integrated by using a fourth order Runge-Kutta method. The integration

time increment is based on the shortest natural period of a system with comparable stiffness parameters. Since the gear meshes have time varying stiffnesses, the system has no discrete natural frequencies. Even if average stiffness values are used, the process to solve for the natural frequencies of a nine degree of freedom system becomes a tedious exercise. Appendix C gives a method for solving the natural frequencies of a PGT based on constant, average mesh stiffness, as well as a technique to set up a simplified model to determine a natural period from which the integration time step may be determined. The time step is taken to be one-tenth of the shortest natural period. At higher input speeds, the mesh period decreases to the point where it is equal to the natural period. At this point, and for all higher input speeds, the integration time step is taken to be one per cent of the mesh stiffness period.

The numerical integration duration is left as a user input variable. The integration should be carried out at least for a length of time sufficient for the start-up transient to decay. The proper length of integration is dictated by how often the synchronous position occurs in the model. As mentioned in Section 3.4.3, if all n planets have either no or identical profile errors, a synchronous positions takes place as often as each new tooth engagement at the reference sun/planet mesh. For these conditions, the integration duration should be the time required for the sun gear to make one complete revolution. If all n planets have

distinctive errors, the synchronous position occurs much less frequently and the length of integration should be adjusted accordingly.

During the entire integration process, results from the static analysis programs are used to determine the following dynamic information:

a. How the individual mesh dynamic loads are shared among the contacting tooth pairs during periods of multiple tooth contact.

b. Variation of the transmission ratio of a contacting tooth pair as the pair moves through the contact zone.

While individual mesh parameters are based on static results, system behavior, such as load sharing among planets and movement of the sun center, is calculated independently and entirely from the differential equations of motion. Consequently, comparing the dynamic results for a quasi-static operating state with the static results should offer a guide as to the validity of the dynamic model and the integration process.

For the dynamic analysis, it is assumed that the loaded mesh arcs are of the same length as those experienced in the static mode. This is a reasonable assumption because the rapidly changing loads should not cause a permanent change of the meshing arc lengths. It is also assumed that the sun center motion has little affect on the mesh stiffness characteristics. This seems acceptable because the magnitude of the sun center displacement is smaller than what Hidaka

[23] and Pintz [7] considered was necessary to appreciably affect the meshing characteristics.

With these assumptions, the dynamic absolute angular displacement of the sun gear is used to interpolate mesh conditions in the static mode for all 2n gear meshes in the PGT. In such a manner, the parameters listed above are used in the interpolations between dynamic and equivalent static positions to calculate the dynamic mesh forces.

The dymanic mesh force represents the amount of load increment due to the periodic excitations within the system. The dynamic load factor is used as a means of measuring this increment based on the nominal static load. It is also used as a penalty to de-rate the load capacity of a PGT based on the transmitted power. There are two ways of calculating the dynamic load factor. One is based on the total dynamic mesh force at each engagement, the other is based on the load sharing that occurs within that mesh.

In the first case, the dynamic load factor at the $j^{\underline{th}}$ position is defined as:

$$DF_{1ij} = \frac{GPTF_i}{Q_j} \qquad i = 1, \ldots, n \tag{3.56}$$

where: $GPTF_j$ = dynamic gear pair transmitted mesh force

Q_j = nominal static transmitted mesh force

The force Q_j is calculated in the static analysis and is determined by the load sharing within the system. DF_{1ij} can be thought of as the dynamic load factor of the individual

gear mesh and its shafts and bearings.

The dynamic load, QD, for the k^{th} contacting gear tooth pair at the j^{th} dynamic mesh position is:

$$Q_{Dijk} = \frac{KP_{ik}}{KP_{ij}} Q_j \quad i = 1, \ldots, n \tag{3.57}$$

where: KP_k = gear tooth pair stiffness

K_{ij} = total mesh stiffness

The second dynamic load factor can be defined as:

$$DF_{2ijk} = \frac{QD_{ijk}}{Q_{ijk}} \tag{3.58}$$

The force Q_{ijk} is the component of the total force Q_{ij} which is transmitted by the k^{th} contacting gear tooth pair. It also is calculated in the static analysis. DF_{2ijk} is the dynamic load factor for an individual gear tooth pair traversing the mesh arc. It is particularly important when the strength of the gear teeth is of primary concern. The larger of the two calculated dynamic load factors will be used as the design dynamic load factor, DF.

3.6 Computer Programs

The development of a set of computer programs which implement the analytical work presented in Sections 3.4 and 3.5 was one of the stated goals of this investigation. These programs constitute a comprehensive, analytical tool enabling the user to quickly evaluate PGT transmission designs. The

programs were written with the intent of running them on a mini-computer system. Memory constraints of mini-computers place a premium on efficient programming techniques, and, the direct inter-active capabilities offer the user much more flexibility in executing these programs than is available in batch mode systems. The computer programs in their entirety, along with sample outputs, are listed in Appendix D. Highlights of the computer programs, their structure and inter-dependence, explanation of data transfer, and results available for output are discussed in this section.

The PGT transmission analysis package is written in Fortran IV for use on a Hewlett Packard HP-1000 F Series computer with the RTE-IVB operating system. The programs have been arranged into three groups which may be classified as the pre-processor, processor, and post-processor. Figure 26 shows the block diagram of the entire package. Each program may be compiled, loaded, and executed independently with its results being stored in data files for either immediate examination, or use by subsequent programs. The parallel development of external-external and external-internal spur gear programs also enables the user to evaluate a single stage parallel-shafted transmission.

3.6.1 Pre-processor

The pre-processor group consists of the programs which do the following tasks:

a. Accepts, and organizes input data, and writes input data to a data file

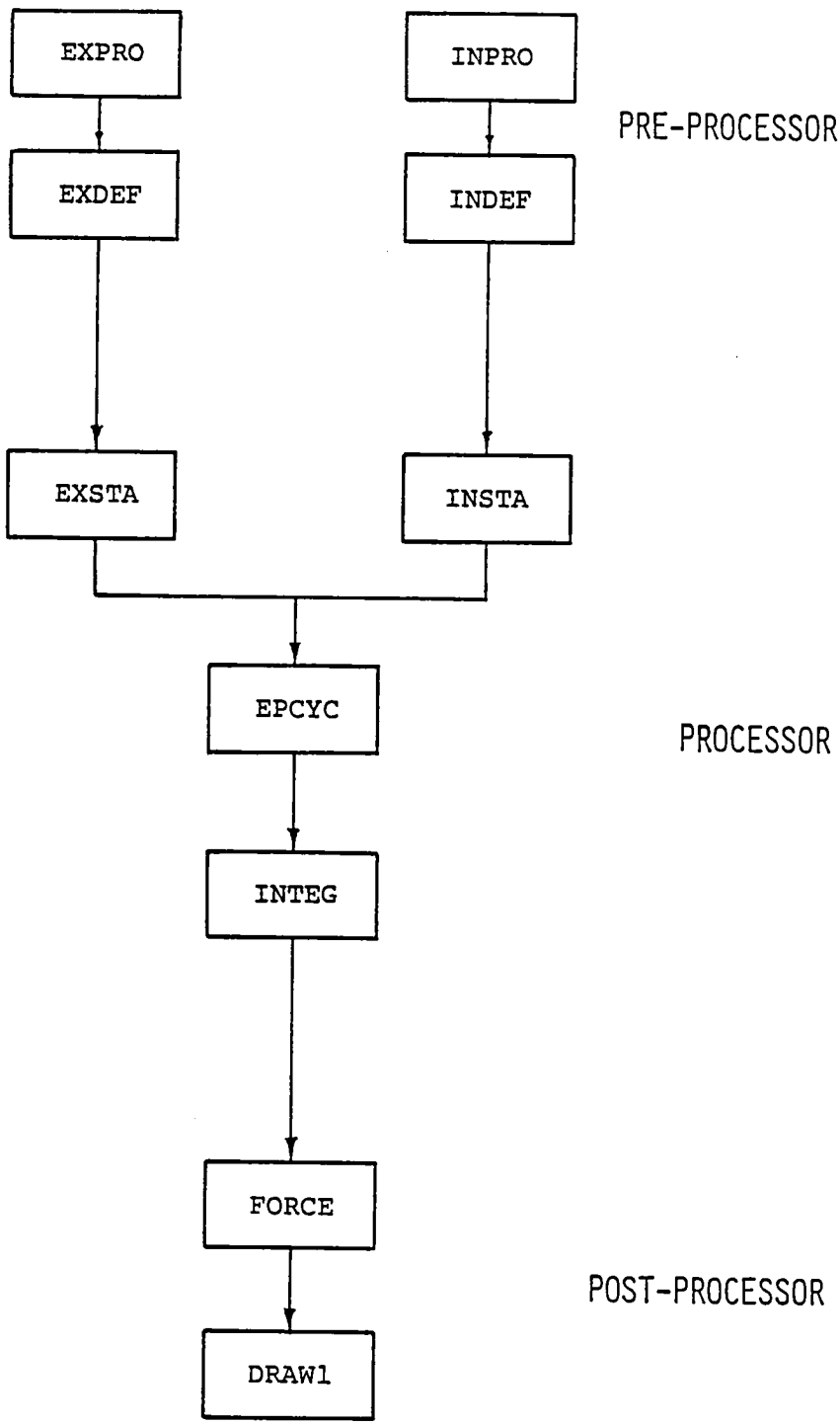


Figure 26 - Block Diagram of Programs

b. Defines tooth profile

c. Determines stiffness characteristics of gear teeth

The programs EXPRO and INPRO define the tooth profiles by digitizing the involute into 100 points defined by a gear based coordinate system. Both standard and non-standard gear forms can be digitized. These programs also check for interference between mating teeth and calculate the theoretical contact ratios. These results are written to data files EXPROD and INPROD The programs EXDEF and INDEF use these results to calculate tooth deflections due to a unit load as it is applied to all the digitized profile points. These results are written to data files EXDEFD and INDEFDT for use in the processor programs.

3.6.2 Processor

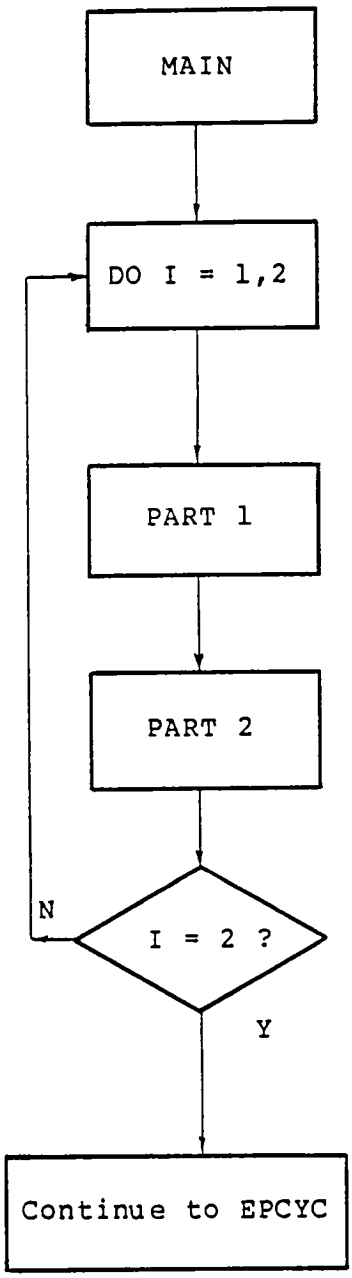
The processor group contains the programs which perform the actual static and dynamic analysis of the PGT. The static analysis for an external-external and an external-internal mesh are done as parallel tasks. These results are then combined based on the configuration of the PGT. The dynamic analysis is then performed taking into account all 2n individual mesh stiffnesses.

The static analysis EXSTA and INSTA use an iterative process to solve the indeterminate problem of static load sharing among multiple tooth gear meshes. First, the gears are treated as perfectly rigid bodies in order to use geometric analysis techniques to establish points of contact

for 50 positions in the mesh arcs. Then the results of EXDEF and INDEF are used to redefine the tooth profiles to reflect the loaded and deflected engagement conditions at 50 positions along the now extended mesh arc. In this manner, it is possible to determine the stiffness of the individual pairs, as well as the variable mesh stiffness as a function of the driving gear's position.

Because of the length of these programs and memory limitations of the HP-1000, it was necessary to use segmentation techniques to enable these programs to be executed (Fig. 27). Since both programs have identical structures, common segment names have been used. Segment MAIN acts as the executive controller for the programs. It reads and stores the profile and deflection data previously calculated. Segment PART1 determines the initial and final positions of contact, establishing the extremities of the mesh arc. Segment PART2 then determines the position of contact and remaining static mesh information for 48 equally spaced positions within the mesh arc. By executing PART1 and PART2 in a two-pass do-loop, it is possible to evaluate first the unloaded, rigid body, and then the loaded and deflected behavior of the gear mesh.

EPCYC assembles the mesh information from EXSTA and INSTA for use in the dynamic analysis. The main task of EPCYC is to properly orientate the mesh stiffness of all the sun/planet and planet/ring engagements to properly show the phase relationships among them. This will allow tracking and



$I = 1_i$ Rigid (unloaded) analysis

$I = 2_i$ Deflected (loaded) analysis

Part 1 - initial and final postiions of contact

Part 2 - static mesh information (contact position, stiffness etc) within the mesh arc.

Figure 27 - Segmentation of EXSTA and INSTA

interpolation of all the mesh characteristics as a function of the sun gear position.

The program INTEG performs the numerical integration of the differential equations of motion given in Section 3.5. The calculations are based on a dynamic cycle which starts with the initiation of contact as a tooth pair enters the contact zone and ends with the initiation of contact with the following tooth pair. INTEG examines the mesh stiffness function for the reference sun/planet mesh to determine the arc length of this cycle. This arc length is then compared against the absolute dynamic angular position of the sun gear to establish the number of sun gear mesh engagement cycles which have been integrated, and also to interpolate the instantaneous position within the mesh arc of the currently engaged sun/planet gear teeth. By tracking the sun gear position this way, it is possible to interpolate mesh stiffness values for all the remaining gear meshes and calculate their instantaneous dynamic forces.

The total number of sun gear mesh cycles to be integrated is user defined andshould bebased on the synchronous positioning of the PGT. As defined in Section 3.4.3, a synchronous position occurs with a duplication of a reference engagement state. After completion of integration of the pre-determined n number of sun gear mesh cycles, the dynamic force data is stored in a buffer, which is then written to a data file. By varying the data sampling rate, the user can adjust the length of integrated time over which

data is stored. In the present version of the program, a buffer length of 6000 elements is used to store the results. This allows the storage of 6000/(m+1) independent, i.e., time, and 6000/(m+1) dependent values for m variables.

The integration of the differential equations is done using a fourth-order Runge-Kutta technique. This is performed by the use of two subroutines in INTEG, RKUTTA and MORERK. The subroutine RKUTTA deeps track of the step size and number of iterations across the integration interval, while MORERK evaluates the derivative being integrated and sums these values across the integration interval. The results of the integration for the displacement and velocity of each mass are the amounts of deviation from the steady-state values over that time interval. These are added to the steady-state values to obtain the absolute displacement and velocity of each element. The dynamic forces are then calculated based on the relative motion of the elements. These values are then stored for examination by the post-processor program.

3.6.3 Post-processor

The function of the post-processor is to provide access to all the data that has been generated by the above programs. The user has the option of presenting this data in either tabular or graphical form. The program which evolved during the course of this investigation has proved particularly useful in evaluating the calculated results and is described in this section.

The program FORCE calculates the static load sharing among the planets and then the tooth load sharing among the gear teeth within the same mesh for a given position of the sun gear. Also, included in the analysis is the position the sun center assumes if it is allowed to move to compensate for unequal planet loading. By evaluating these forces and movements at a specified number of discrete positions within one sun gear's engagement cycle, a baseline is determined against which the dynamic forces and movement can be compared. This results in the calculation of the dynamic load factors as a function of sun gear position. The user can direct this output either to the terminal screen for a quick indication of trends, periodicity, and maximum and minimum forces; or to the printer for more accurate examination of the results. In addition, the locus of the sun center static and dynamic movement may be plotted.

The program DRAW can be used to graphically examine the integration results. The user specifies the data file to be read and offers the option of graphing the sun/planet mesh stiffnesses and dynamic forces, the planet/ring mesh stiffnesses and dynamic forces, and the dynamic sun gear center displacement. Output can be directed to either the screen or four pen plotter.

CHAPTER IV

RESULTS, DISCUSSION AND SUMMARY

4.1 Results and Discussion

4.1.1 Introduction

The planetary gear train (PGT) analysis methodology developed for this thesis was used to perform a series of comparative parametric studies in order to validate the methodology and assess the PGT drive performance. Design parameters of the gears and system properties used in this study are given in Table 2. The analyses of this particular PGT considered tooth profiles, sun support stiffness, critical damping ratios, shaft stiffness, and transmitted power. Static and dynamic computer analyses were conducted under identical conditions.

4.1.2 Static Analysis

The equivalent gear train of the model shown in Figure 25 was used in the static analysis to investigate the planet load sharing and displacement of the sun gear. Figure 28 shows the phasing and magnitude of the stiffness functions used for the static analysis. Two conditions were examined; a floating or free, sun gear, and a fixed sun gear. The sun gear fixity was determined by its support stiffness. The floating sun gear was allowed to translate in the x and y directions by setting the two perpendicular sun support stiffnesses to 1.0 lb/in. Fixed sun gear conditions were

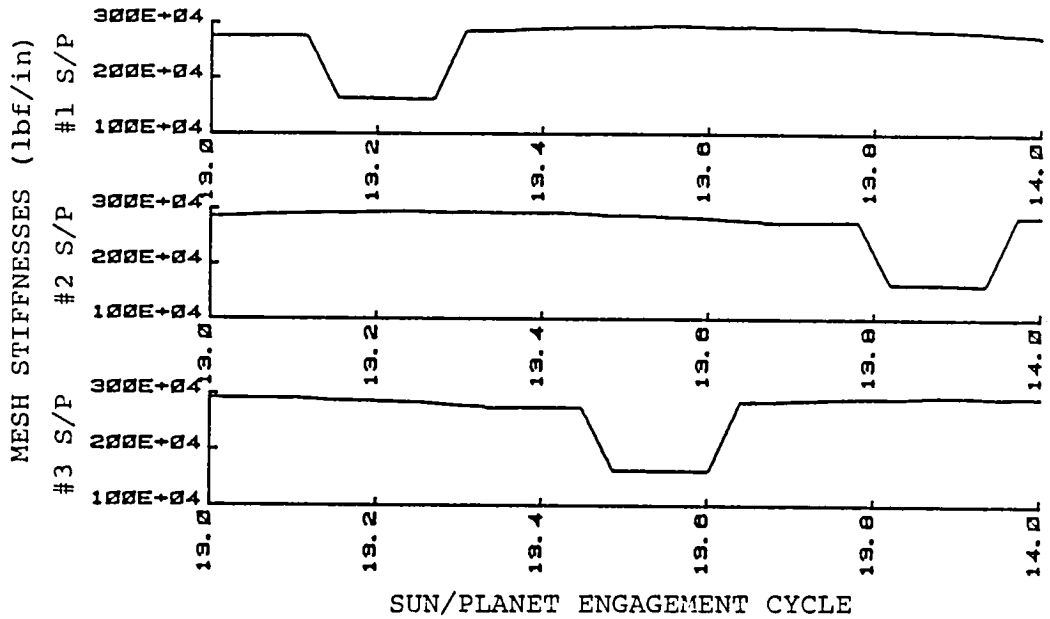
Design Parameters

Diametral Pitch	5
Pressure Angle	22.5^{o}
Number of Teeth	
Sun	25
Planet	24
Ring	73
Face Width	1 in
Addendum	0.20 in (1.0/P)
Dedendum	0.27 in (1.35/P)
Theoretical Contact Ratio	
Sun/Planet	1.56
Planet/Ring	1.78

System Properties

J_{Driver}	5.000 in-lb. sec^2
J_{Sun}	0.040 in-lb. sec^2
M_{Sun}	0.015 lb. sec^2/in
J_{Planet}	0.033 in-lb. sec^2
J_{Ring}	2.000 in-lb. sec^2
J_{Load}	5.000 in-lb. sec^2

Table 2 - Design and System Parameters



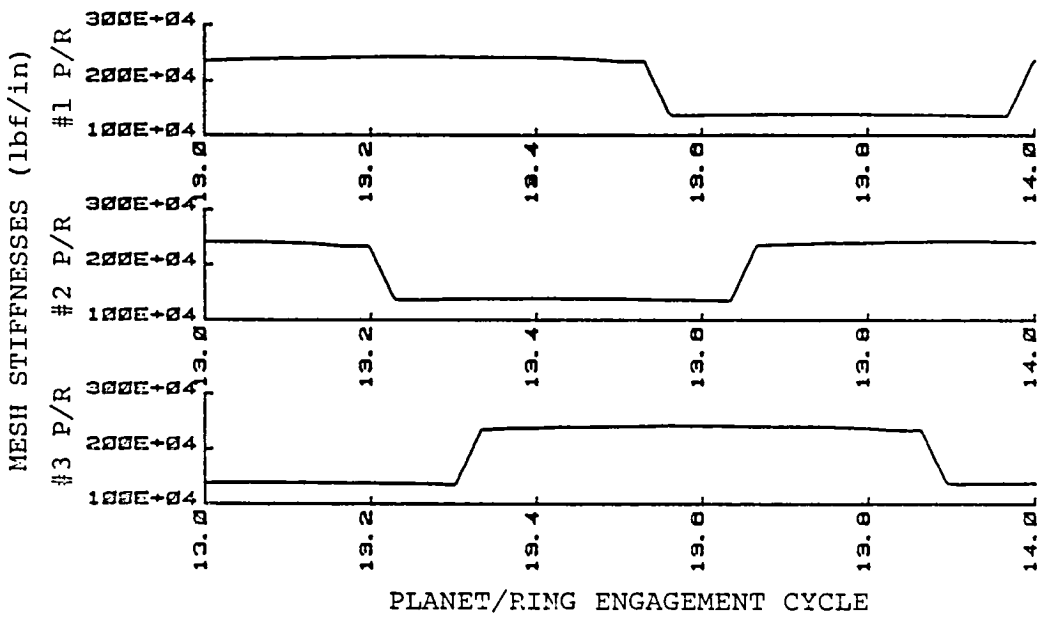


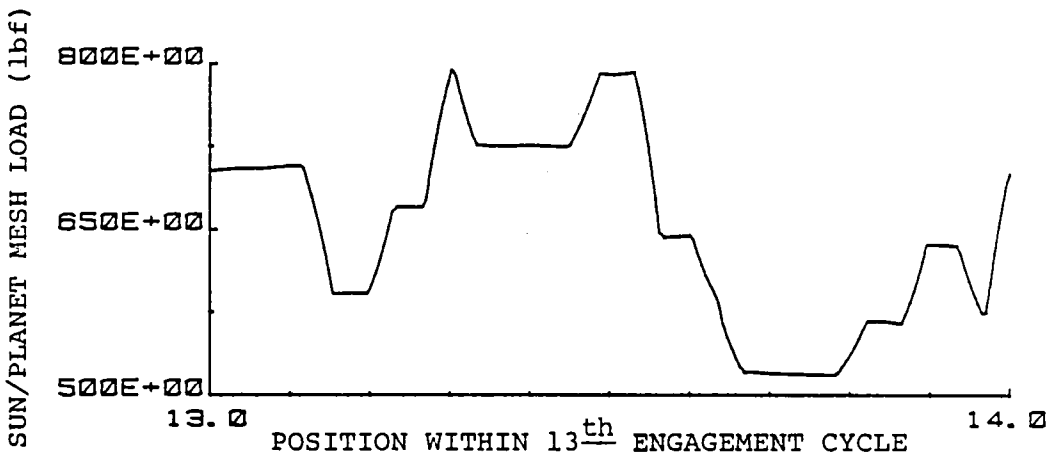
Figure 28 - Phasing and Magnitude
of Stiffness Functions

achieved by assigning a stiffness two orders of magnitude larger than the gear mesh stiffness to the sun gear supports. The results of the static analysis were used as baseline data against which the dynamic results were compared. The torsional stiffness of the input shaft was the same in both cases.

4.1.2.1 Planet Load Sharing

Figure 29 shows the loading for one sun/planet/ring through one sun gear engagement cycle for the fixed sun gear condition. Since the planets are in static equilibrium, the sun/planet and planet/ring mesh loads per planet are equal, The loads illustrate the phasing of the mesh stiffnesses about the sun gear. These static results were obtained by evaluating Equations 3.25 and 3.26 for 100 intervals during the engagement cycle. These results agree with Hidaka's approach [21] who treated the sun/planet and planet/ring stiffnesses as springs in series and then proportioned the sun loads according to the relative stiffness of the equivalent spring. However, Hidaka took the three sun/planet stiffnesses and three planet/ring stiffnesses to be equal. Therefore his load distribution does not reflect the VVMS at each mesh which causes load variations during the engagement cycle.

Figure 30 shows the load sharing among the engaged teeth for one mesh cycle at one sun/planet/ring interface. As the engagement state changes from two to one tooth pair contact, the total load generally decreases. Changes in the



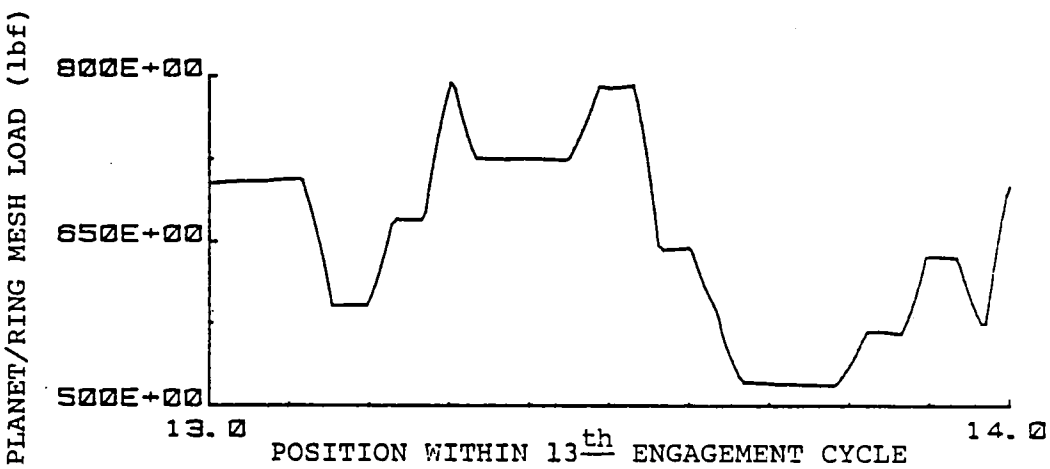
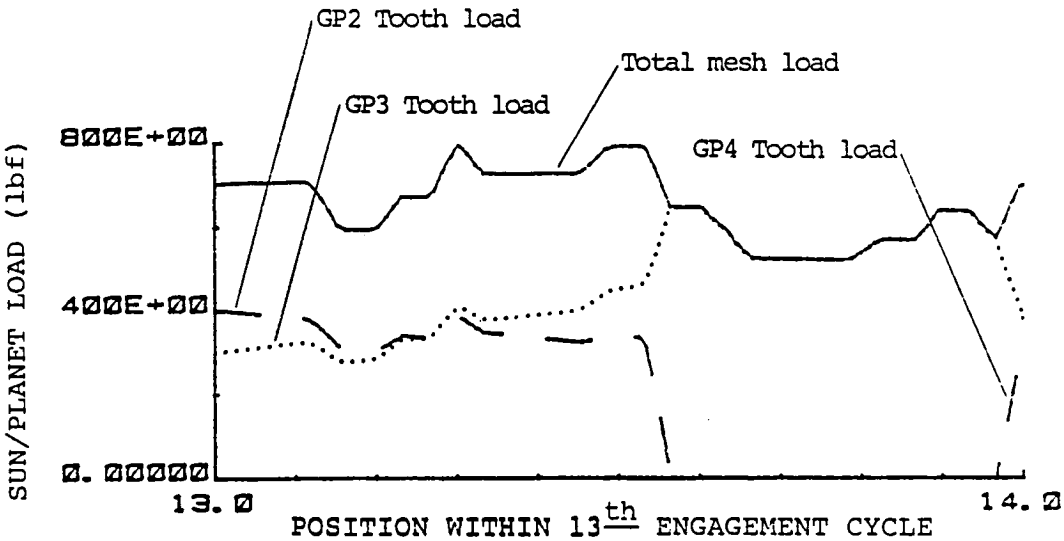


Figure 29 - Load Distribution for One Planet



$K_{SUN} = 1 \times 10^{8}$ lbf/in

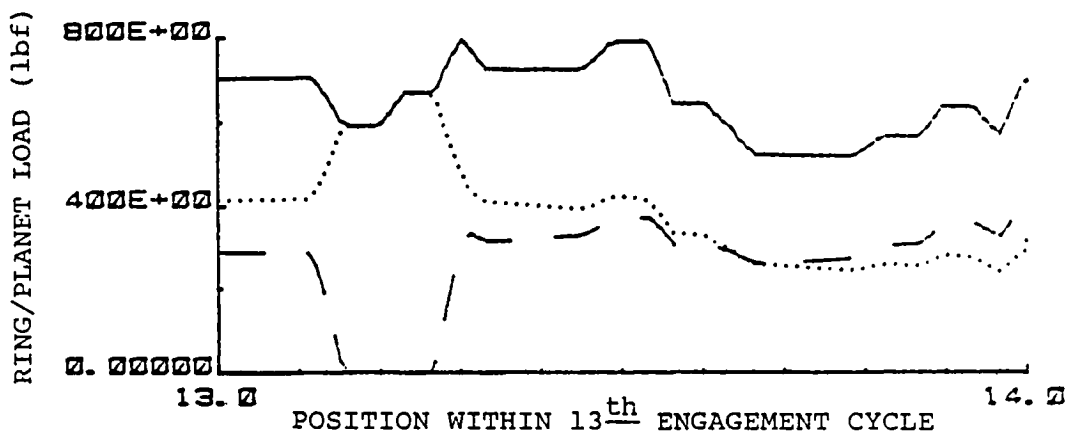


Figure 30 - Load Sharing among Engaged Teeth

sun/planet contact state affect the planet load more noticeably than changes in the planet/ring contact state.

Figures 31a,b,c show that as the sun support stiffness is reduced, the planet load becomes a constant value independent of the sun's position, implying equal load sharing. Table 3 gives the ranges of both mesh loads and tooth loads as the sun gear support stiffness is reduced. Thus, for static conditions, equal load sharing can be achieved by allowing the sun gear to float.

4.1.2.2 Displacement of the Floating Sun Gear Center

As explained in Section 3.4.2.1, the contact conditions at each sun/planet mesh will differ due to the following reasons:

a. The actual number of teeth in contact differ at each mesh.

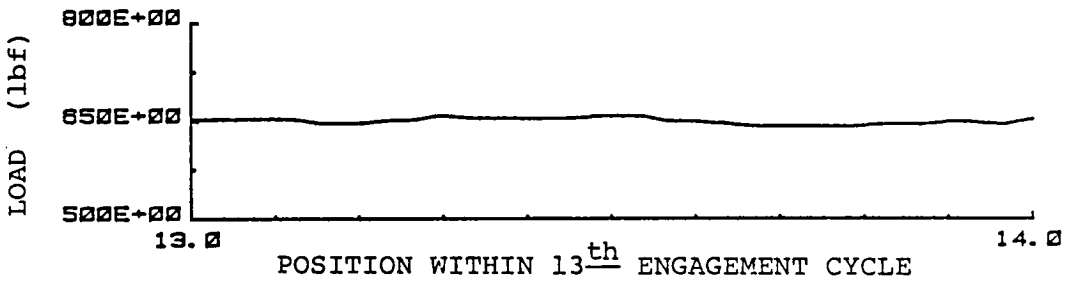
b. The location of contact on the engaged teeth differ at each mesh.

In accordance to the VVMS model, gear mesh stiffness values will also vary at each sun/planet and planet/ring engagement. Consequently, at static load conditions, the sun gear will move to the position such that:

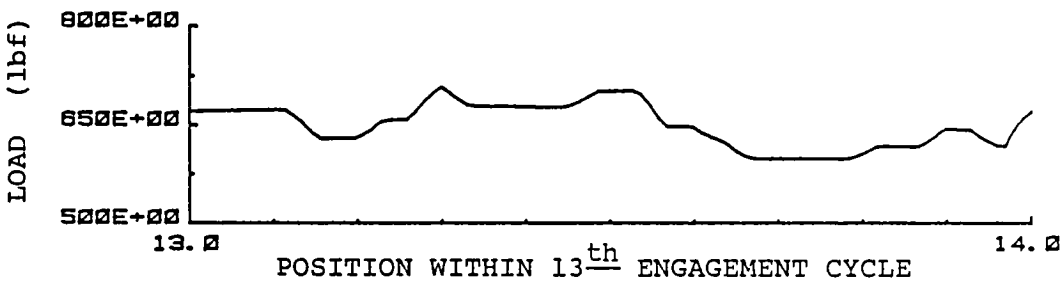
a. The mesh loads collectively produce a torque about the sun's center equal to the input torque.

b. The sum of the mesh loads resolved in the x and y directions is zero.

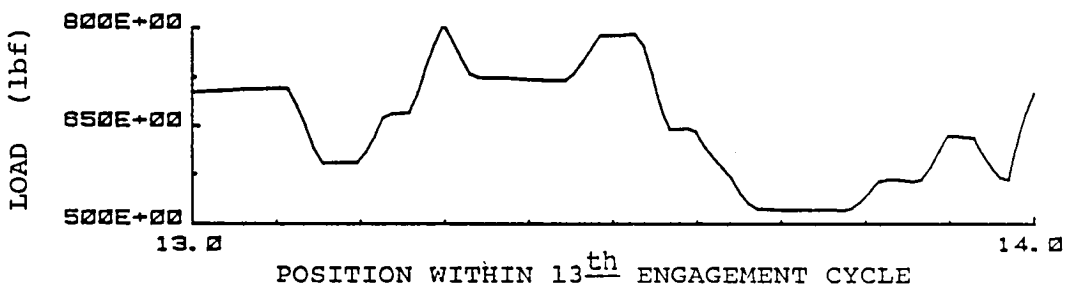
Since the gear mesh stiffness is a periodic function with a period of one pitch, the center of the sun gear is



K_{SUN} = 1 lbf/in



$K_{SUN} = 1 \times 10^{6}$ lbf/in



$K_{SUN} = 1 \times 10^{8}$ lbf/in

Figure 31 - Effect of Sun Support Stiffness on Planet Load

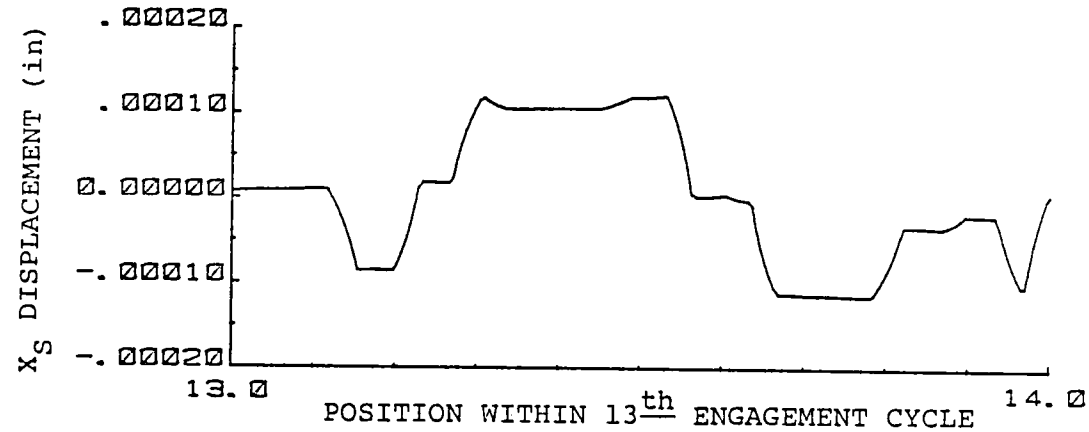
K_{Sun} (lbf/in)	Max Mesh Load (lbf)	Min Mesh Load (lbf)
1×10^8	792	528
1×10^6	753	398
1×10^5	703	610
1×10^4	651	647

Table 3 - Effects of Sun Gear Support Stiffness on Mesh Loads

forced to move with a period of one pitch according to the variation of all the mesh stiffnesses. Since the mesh stiffnesses are implicitly a function of sun gear rotation, the movement of the sun center can be tracked through the sun's angular displacement.

Figure 32 shows the x and y displacement of the sun's center versus the sun gear rotation through one mesh cycle. These curves depict the adjustment of the sun gear position to the changes in the six stiffness functions in Figure 28 in order to preserve the static equilibrium conditions. The displacement curves display two types of motion, dwell and transitional. The dwell motion denotes a stationary position of the sun gear center as the gear rotates, meaning the number of teeth in contact at all six meshes remain constant. The transitional motion is relatively large lateral displacement which occurs during a short interval of rotation of the sun gear. This is caused by a change in the number of teeth in contact at one of the six meshes. For the case of the three planet PGT studied, one mesh cycle had 12 transitions in the number of teeth in contact. Figure 33 is a polar plot of the movement of the sun center. Point 1 is representative of a dwell region as shown in Figure 32. Segment AB is representative of the transitional zone in Figure 32 . Since three identical planets are used, the polar plot also exhibits three-lobed symmetry which is caused by the plasing of the mesh stiffnesses among the planets.

4.1.3 Dynamic Analysis



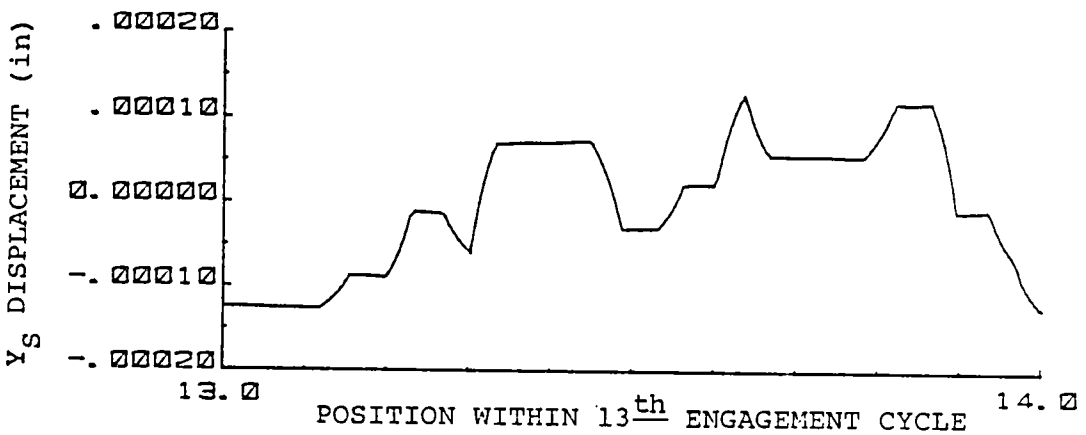


Figure 32 - X and Y Displacement of Sun Gear Center

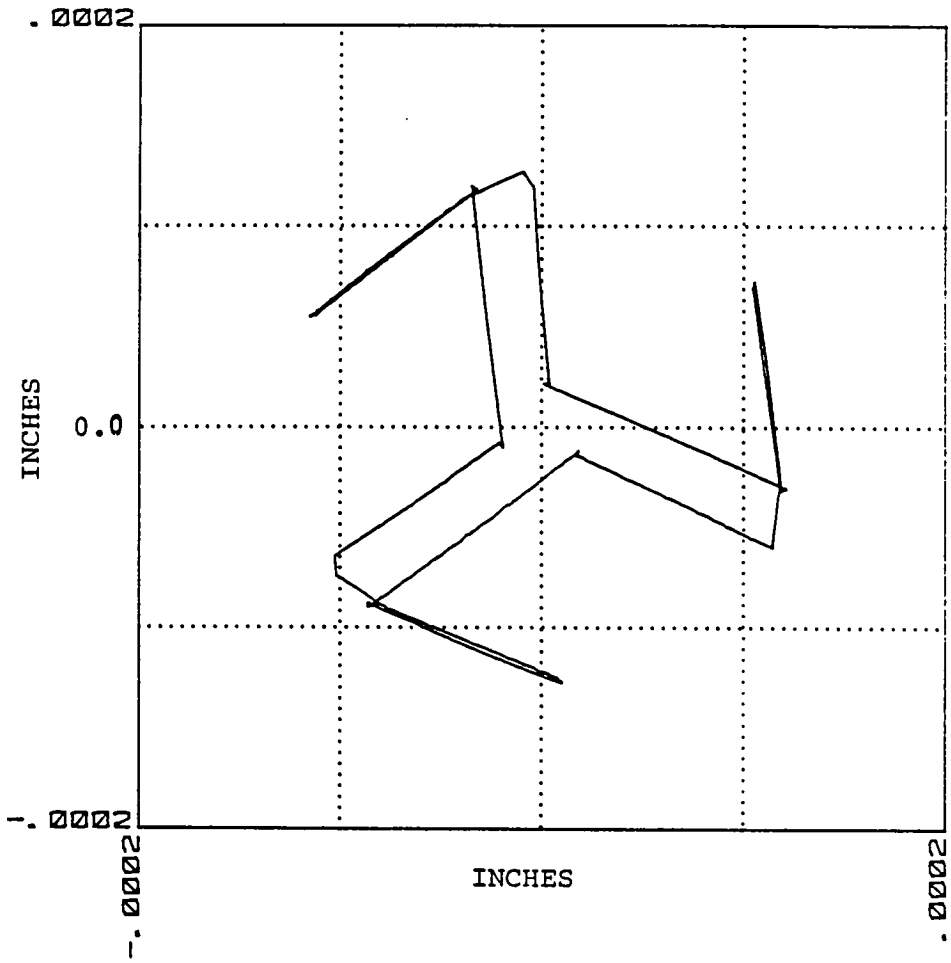
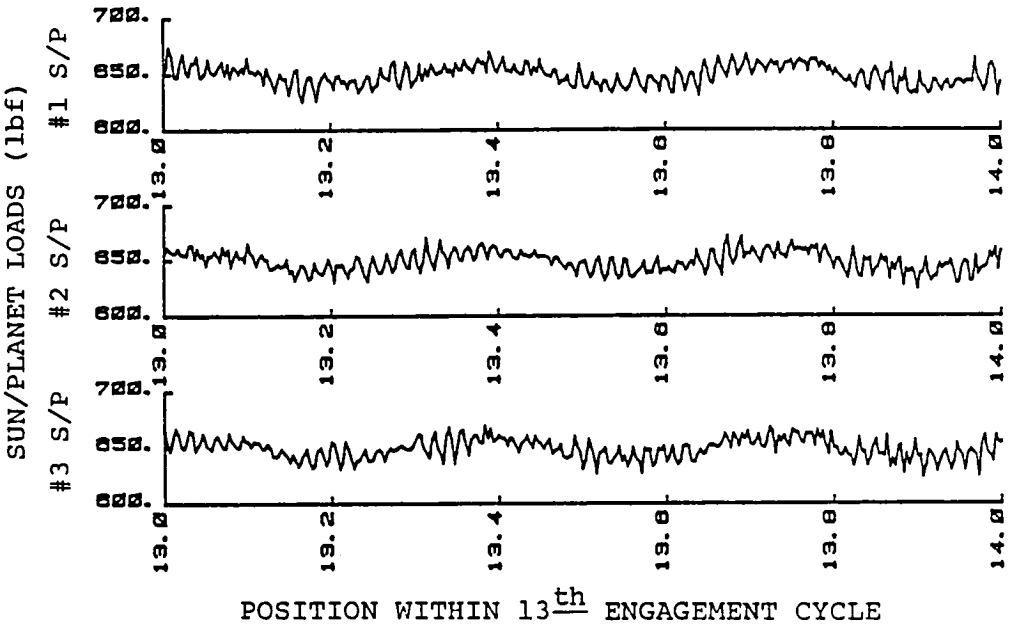


Figure 33 - Polar Plot of Sun Gear Center Movement

The gear train in Figure 25 was used for dynamic parametric studies over an input speed range of 80 to 16,000 RPM. Based on the equivalent gear train concept, the equivalent mesh frequency varied from 155 to 30,000 Hz. The solution of the dynamic equations of motion (Equations 3.36 to 3.43) leads to dynamic loads which depend on the shaft stiffness, operational contact ratio, sun/planet and planet/ring gear mesh characteristics, transmitted loads, damping in the system, and operating speeds. Numerical integration of the differential equations was done on an HP-1000 mini-computer. Stability for the integration process was obtained by using a time step that was five per cent of the system's shortest natural period. The integration was done for a time length equal to 10 times the longest natural period to assure that the start-up transience had decayed to the point where the system was in steady-state operating mode. Appendix C contains a description of the algorithm used to determine these natural periods as well.

4.1.3.1 Low Speed Operation

For low speed operation below the first resonance, it was found that the floating sun gear does position itself to provide for equal loading among the planets (Fig 34). For a static mesh load of 650 lb., (4500 lb-in at the sun gear) the maximum load increment factor was 1.04. The floating sun gear motion was able to compensate for the unequal mesh stiffnesses at each engagement and provide for relatively constant planet loading. This corroborated Hidaka's experi-



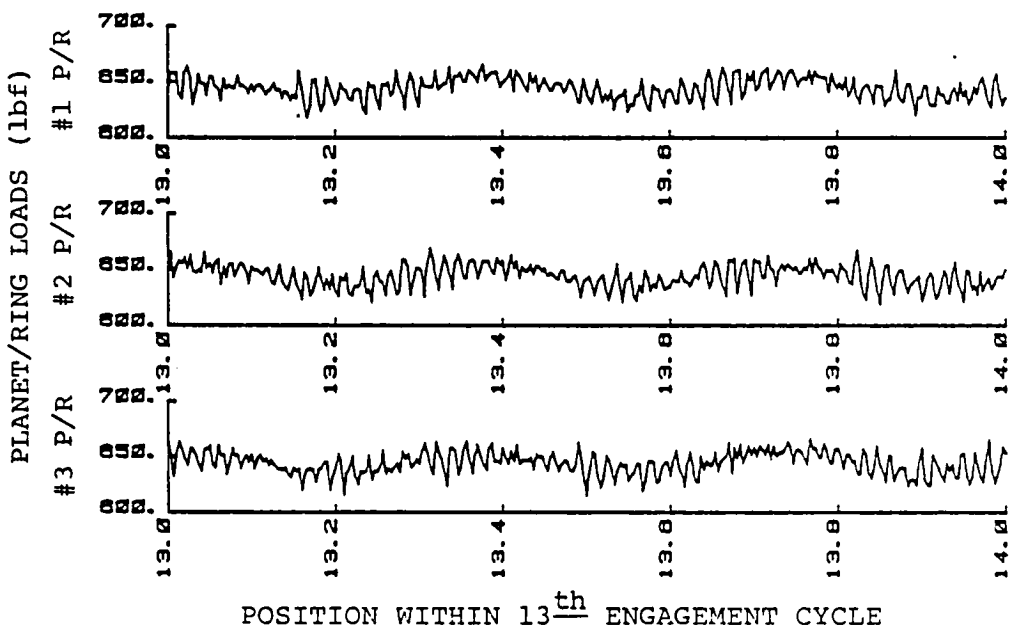


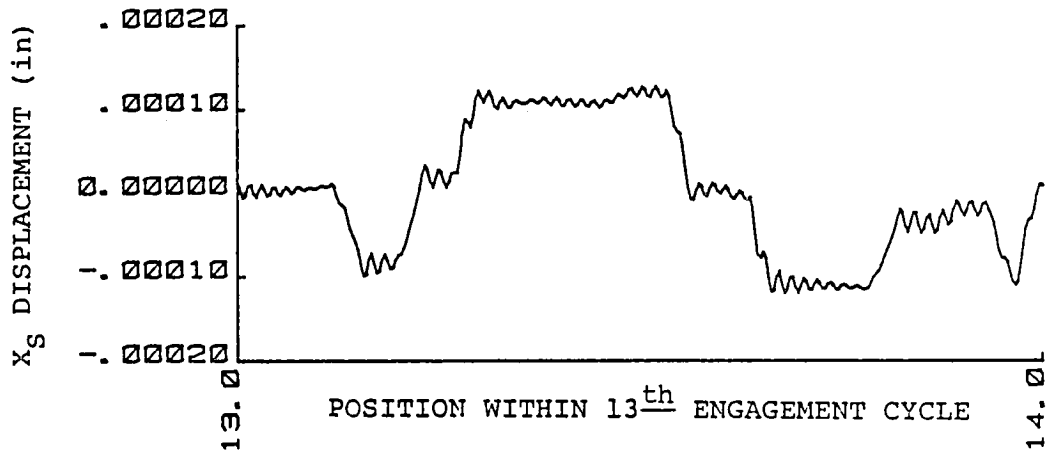
Figure 34 - Planet Loading at Pseudo-Static Conditions

mental results of equalization of load distribution at very low driving speeds. At low speeds, the movement of the sun gear calculated by solving the dynamic equations of motion duplicated the static analysis results (Fig 35). The polar representation of the dynamic x and y translation of the sun gear center better illustrates the dwell region and transitional motion as discussed in Section 4.1.2.2 (Figure 36). Figure 37 shows the mesh loads for a typical planet at low speeds with a fixed sun gear. The extra vibratory phenomena exhibited by the planet/ring mesh is the result of transmission error being compounded through the sun/planet and planet/ring engagements.

4.1.3.2 High Speed Operation

Conditions of operation for the parametric tests of the dymanic model are given in Table 4. Results of the tests for both floating and fixed sun gear conditions over a range of input speeds are given in Figures 38 through 47. The results support Cunliffe's skepticism [14] about the ability of a floating sun gear to compensate for load imbalances and reduce dynamic loading. Equal load sharing among the planets was not preserved at speeds above pseudo-static conditions, and in general, the floating sun gear arrangement lead to higher dynamic loading than a fixed sun gear arrangement under identical operating conditions.

Figures 38 and 39 show the sun gear center locus over a range of operating speeds. At a very low input speed, the sun center follows the regular geometrical motion as



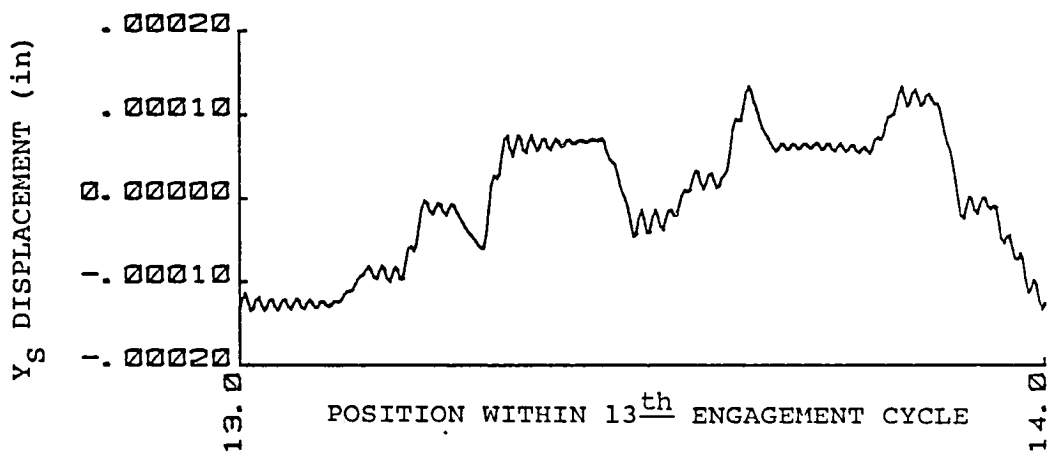


Figure 35 - Movement of Sun Gear Center
at Pseudo-Static Conditions

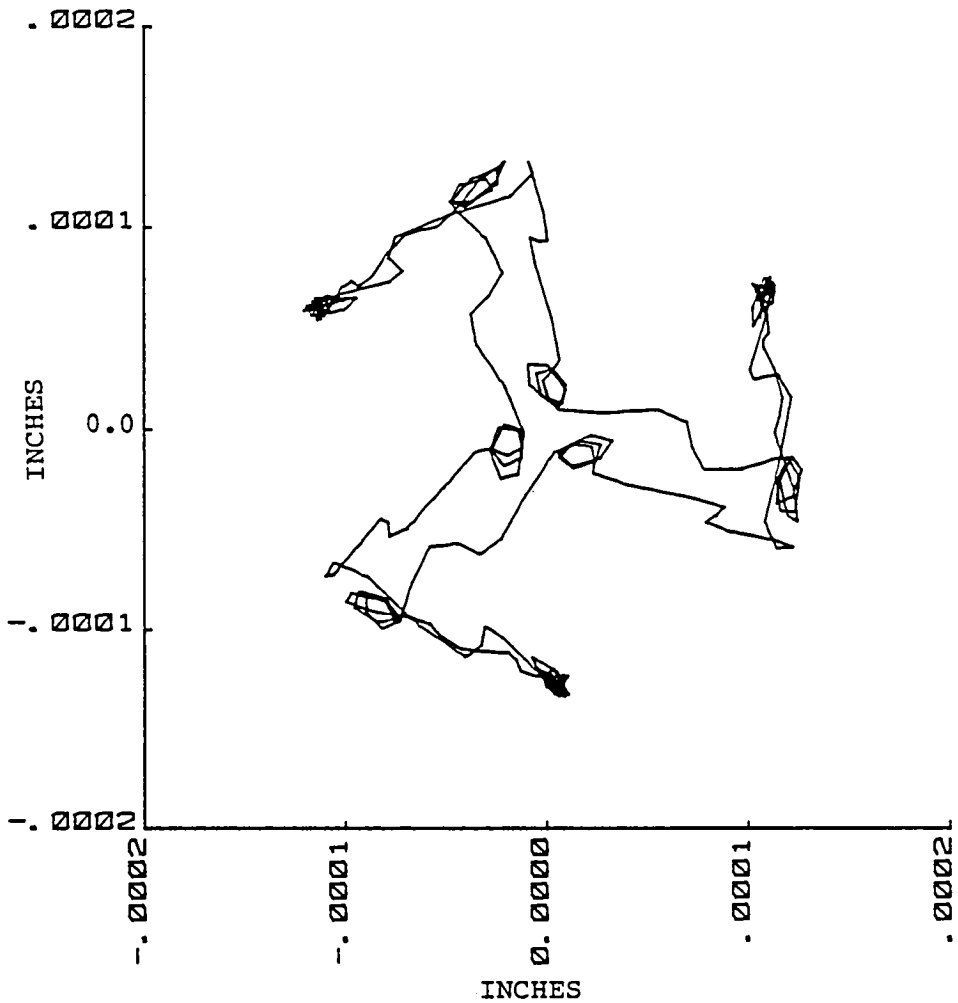


Figure 36 - Polar Plot of Sun Gear Movement
at Pseudo-Static Conditions

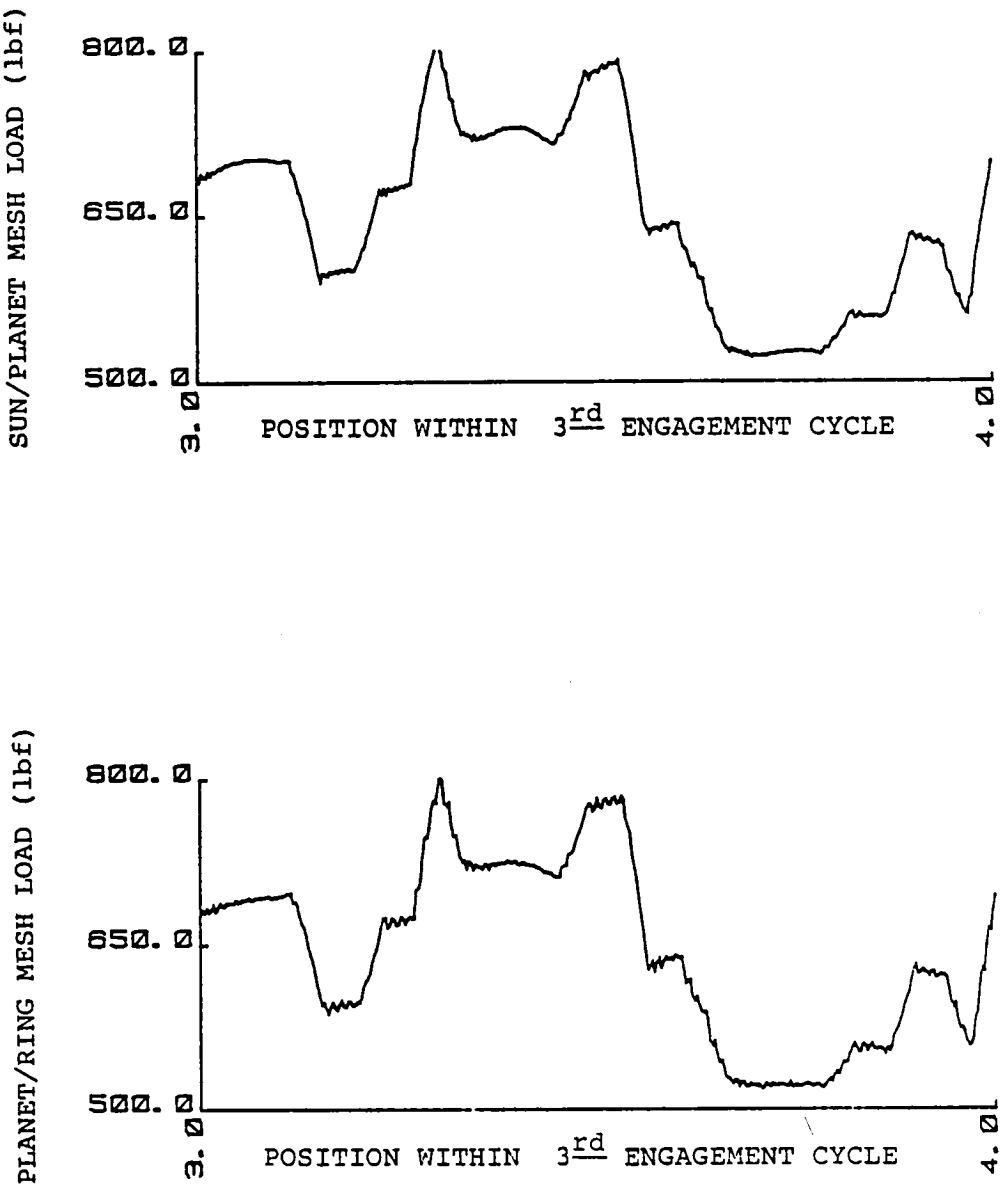
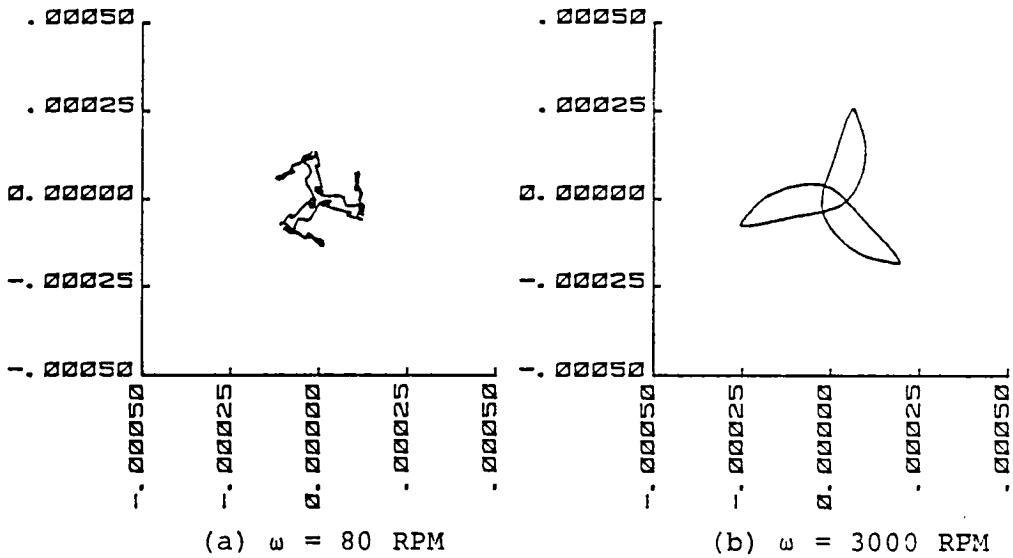


Figure 37 - Fixed Sun Gear Mesh Loads
at Pseudo-Static Conditions

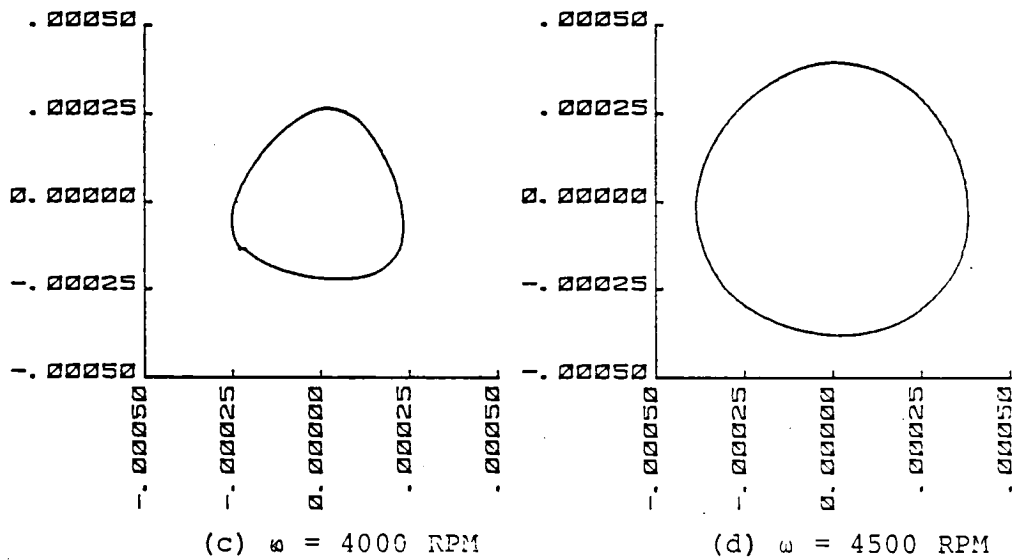
Figure	K_{sun} (lbf/in)	Gear Error (in)	Shaft Stiffness (lbf-in/rad)	Input Torque (lbf/in)
40	1	0	885000	4500
41	1×10^8	0	885000	4500
42	1	2.5×10^{-4}	885000	4500
43	1×10^8	2.5×10^{-4}	885000	4500
44	1	0	185000	4500
45	1×10^8	0	185000	4500
46	1	0	885000	5400
47	1×10^8	0	885000	5400

Table 4 - Operating Conditions



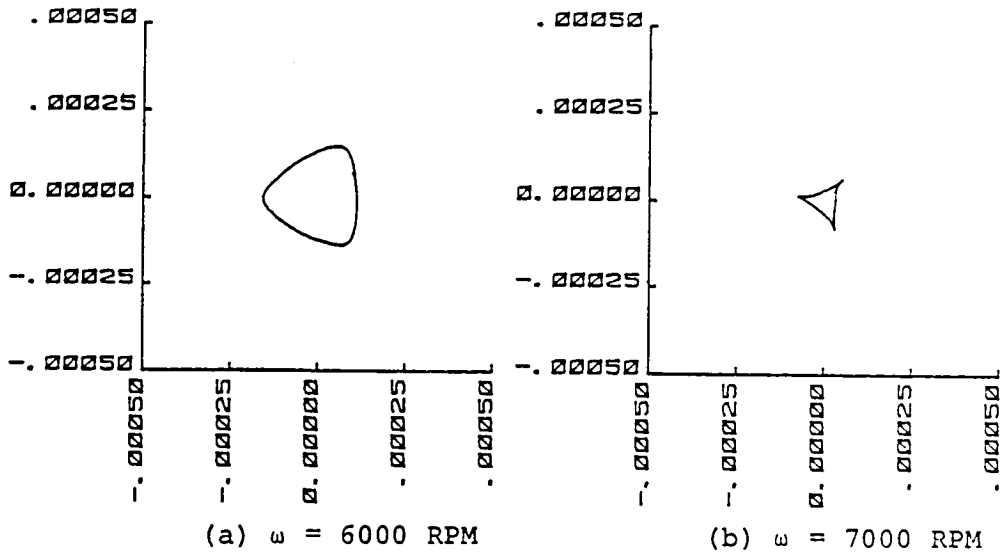
(a) ω = 80 RPM (b) ω = 3000 RPM

(all units are in inches)



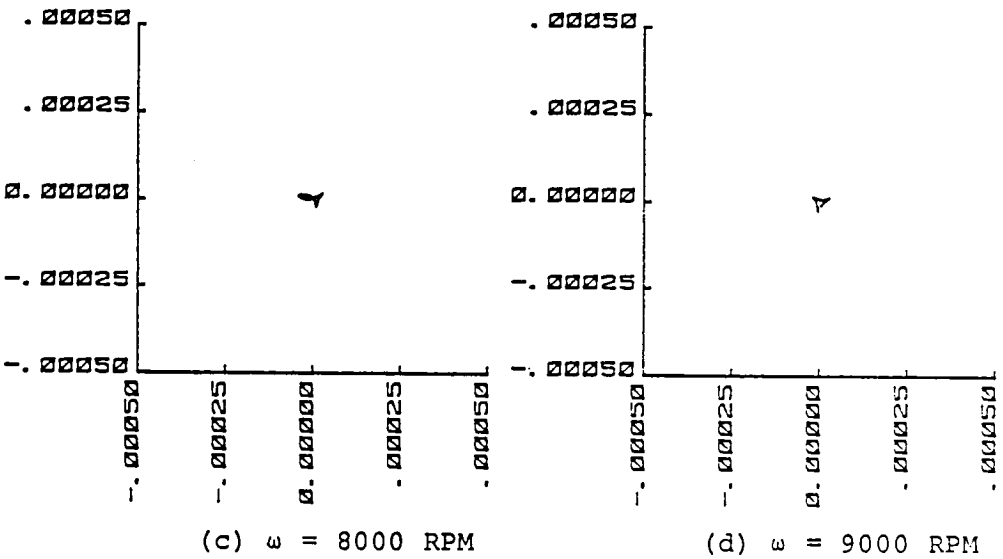
(c) ω = 4000 RPM (d) ω = 4500 RPM

Figure 38 - Sun Gear Center Locus below Resonance Speed



(a) ω = 6000 RPM (b) ω = 7000 RPM

(all units are in inches)



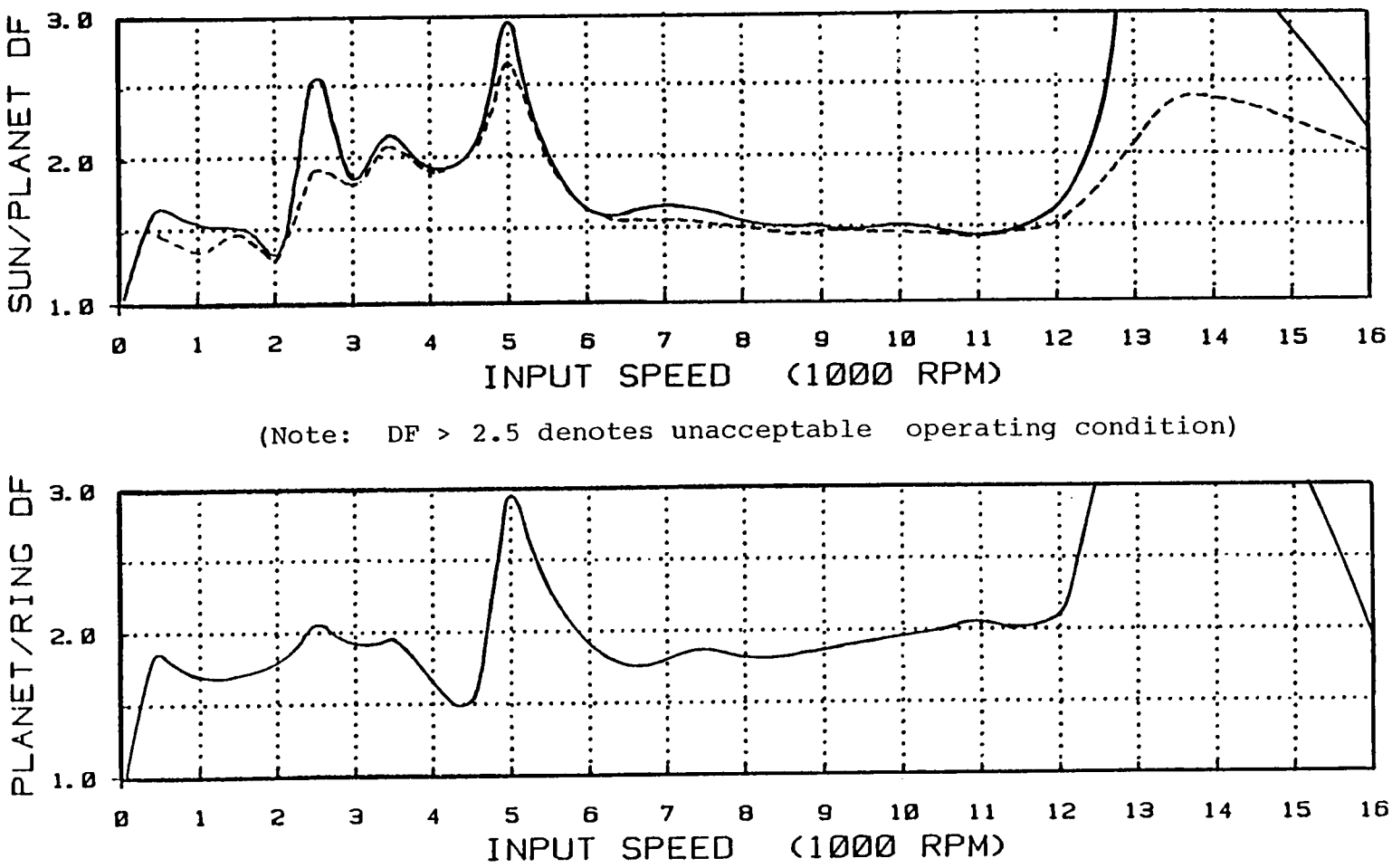
(c) ω = 8000 RPM (d) ω = 9000 RPM

Figure 39 - Sun Gear Center Locus above Resonance Speed

prescribed by the static equilibrium conditions (Fig. 38a). At higher speeds up to the first system resonance, the motion of the sun gear becomes a three-lobed orbit that gradually approaches a circular shape with increasing input speeds (Figs. 38b, 38c, 38d). At speeds above the first resonance, the translational motion of the sun gear decreases, although it still maintains the symmetrical orbit (Figs. 39a, 39b, 39c, 39d). Therefore the influence of the floating sun gear decreases at increasing mesh stiffness frequencies.

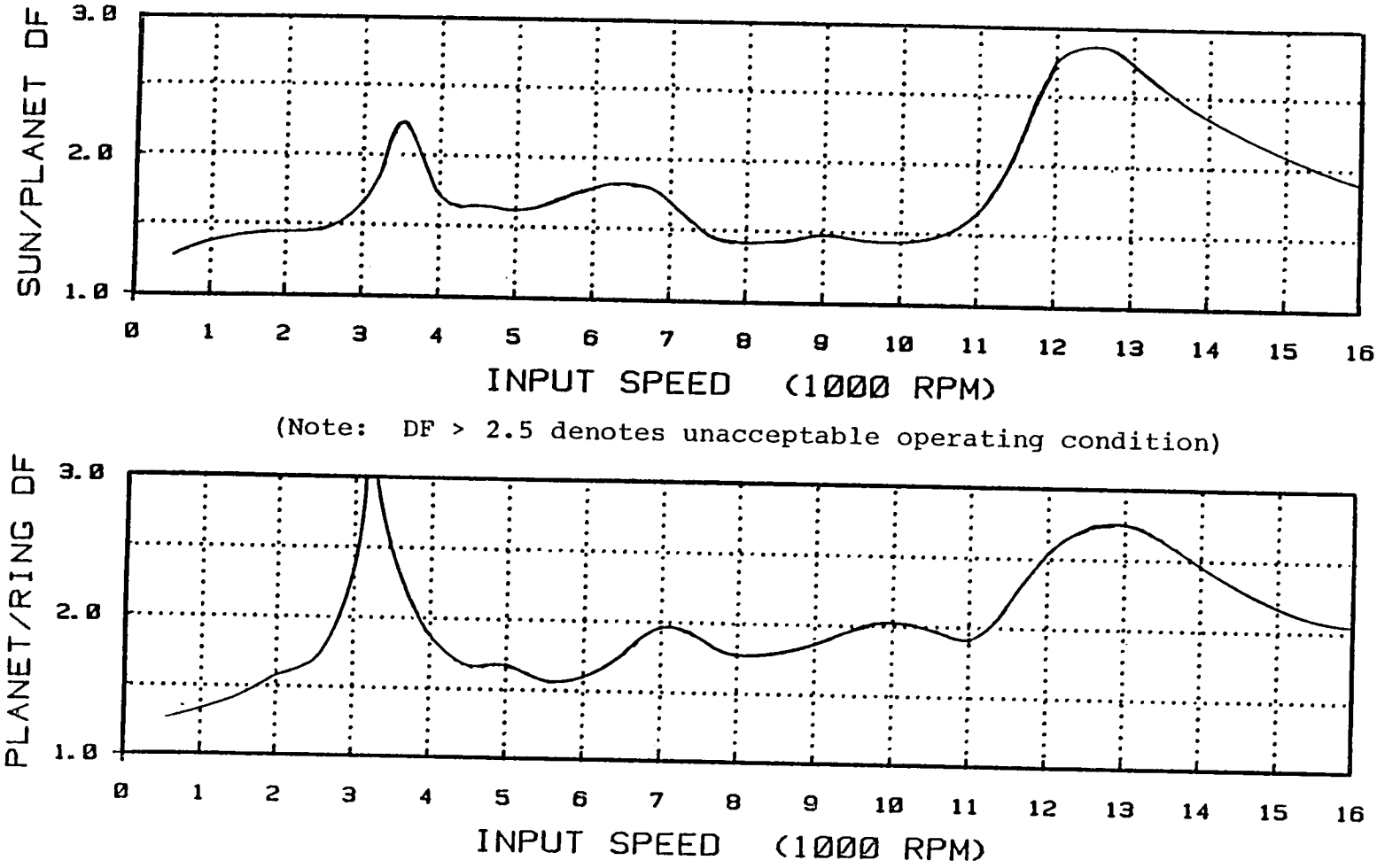
Figures 40 and 41 compare the dynamic factors for the floating and fixed sun gear conditions at the sun/planet and planet/ring meshes. The higher loads shown in Figure 39 in the lower speeds can be directly attributed to the increasing sun gear radial displacement. In non-resonant regions between 6,000 and 11,000 RPM, the floating sun gear arrangement shows generally slightly higher dynamic loads, which are due to the sun gear displacement. Both configuations showed higher dynamic loading at the planet/ring mesh for a given input speed due to the compounded transmission error occurring through the sun/planet/ring meshes. Increased system damping decreased the dynamic loads in Figure 39.

Both figures show similar resonance regions at high input speeds. This is the tooth mesh resonance which produces the highest dynamics in the system. Although the fixed sun gear shows a lower dynamic factor, it was found that a dynamic factor over 2.5 was indicative of tooth separation within the mesh, and therefore, was an unaccepta-



(Note: DF > 2.5 denotes unacceptable operating condition)

Figure 40 - Dynamic Factors for Floating Sun Gear Configuration



(Note: DF > 2.5 denotes unacceptable operating condition)

Figure 41 - Dynamic Factors for Fixed Sun Gear Configuration

ble operating condition. Both fixed and floating sun gears had experienced dynamic factors over 2.5, indicating neither configuration sucessfully minimized dynamic loading at high frequency resonances.

These findings are similar to Cunliffe's [14] who found high dynamic loads at low frequenacies for a floating sun gear and even higher ones at the high frequency resonances. Hidaka's [23] experimental results also show two resonance regions with the highest dynamic loads occuring at the high frequency resonance.

Lanchester, Love, and Allen [8], [10], [29], all pointed out the need for high precision gearing in order for proper operation to the PGT. Figures 42 and 43 show the effect of a sine error on the sun's gear teeth on the dynamic loads. The magnitude of the sine error was purposely set at a relatively large error (0.00025 in.) to illustrate a worse case condition. Independent of the sun gear's fixity, the tooth mesh resonance was particulary sensitive to the gear tooth quality.

Figures 44 and 45 show the effect of having softer input and output shafts. The torsional shaft stiffnesses affect the two PGT configurations in opposite ways. For the floating sun gear, the peak loads were relatively unchanged at the low speed resonance, but were reduced at the high speed range. For the fixed sun gear, peak dynamic loads were reduced in the low speed range, but only slightly decreased at the high speeds.

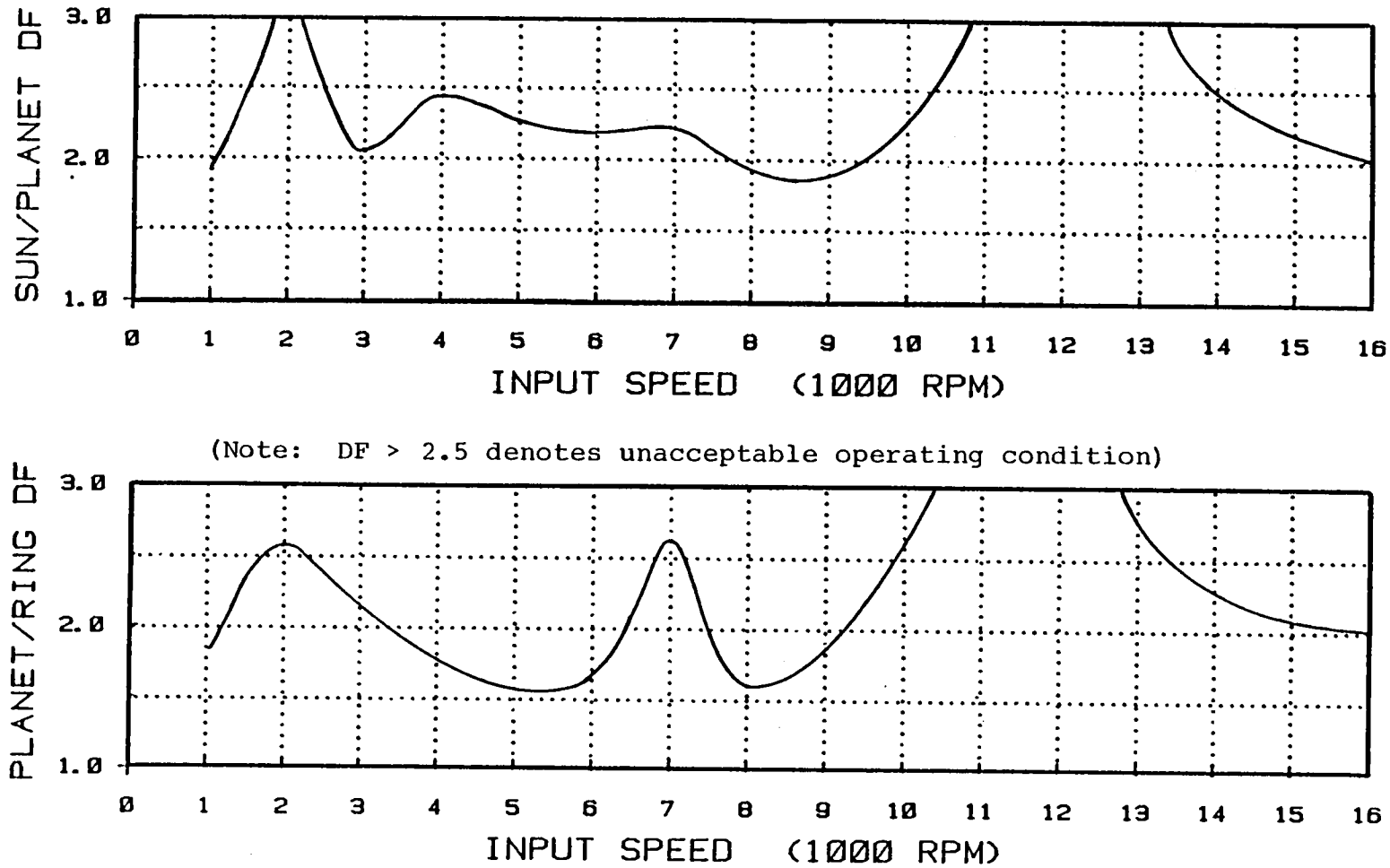
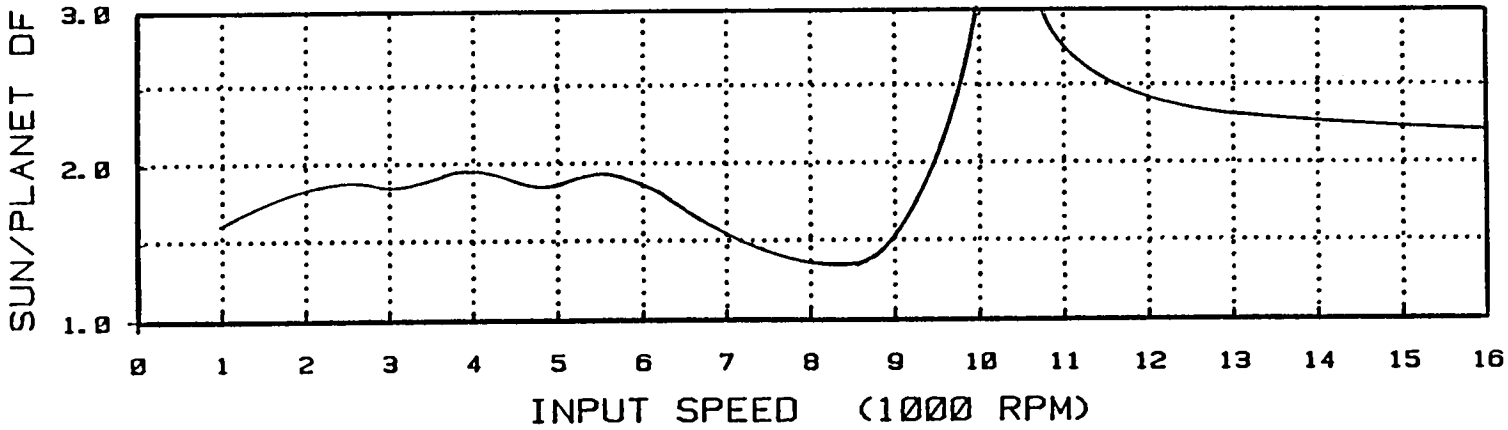


Figure 42 - Dynamic Factors for Floating Sun Gear Configuration with Sine Error



(Note: DF > 2.5 denotes unacceptable operating condition)

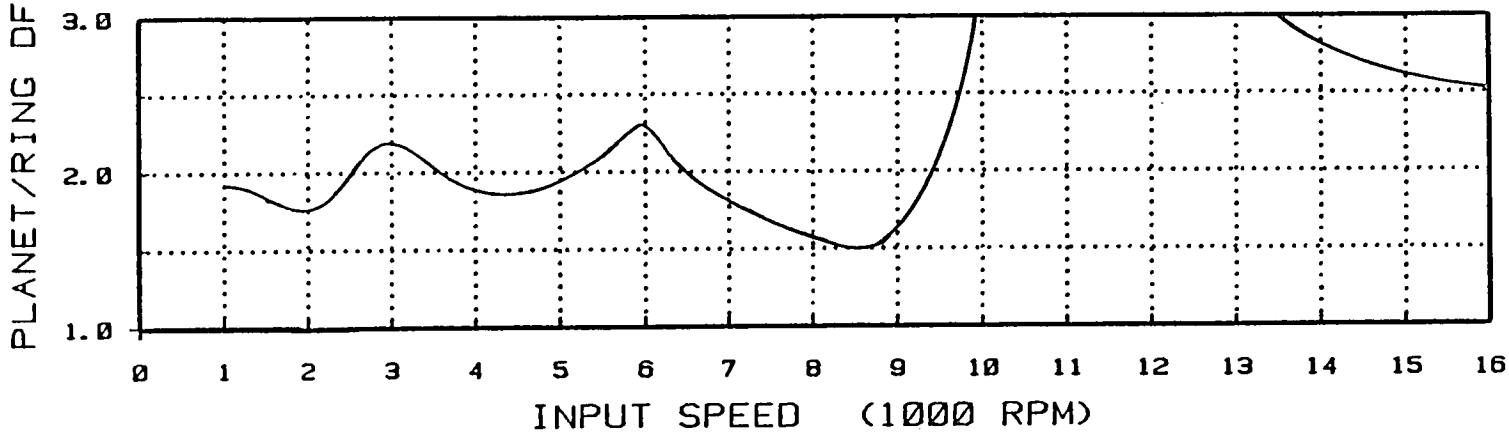
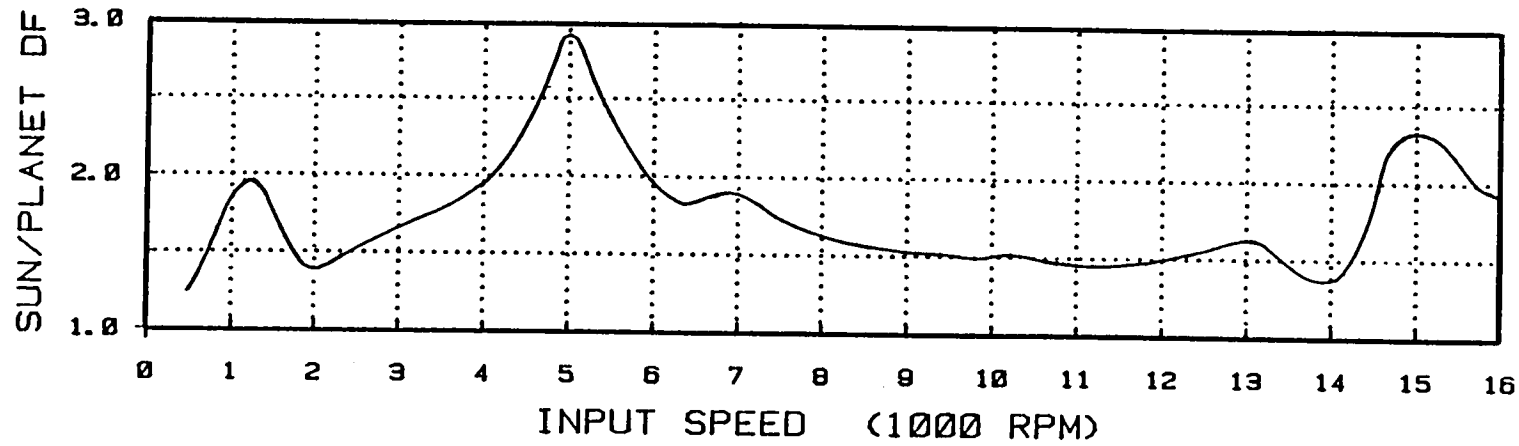


Figure 43 - Dynamic Factors for Fixed Sun Gear Configuration with Sine Error



(Note: DF > 2.5 denotes unacceptable operating condition)

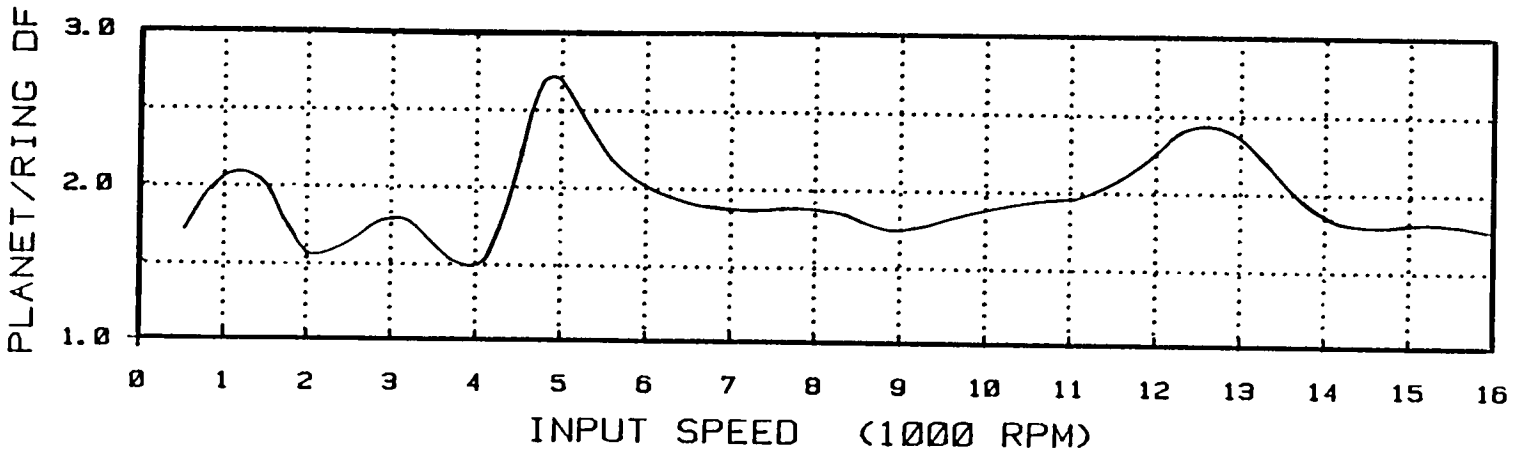
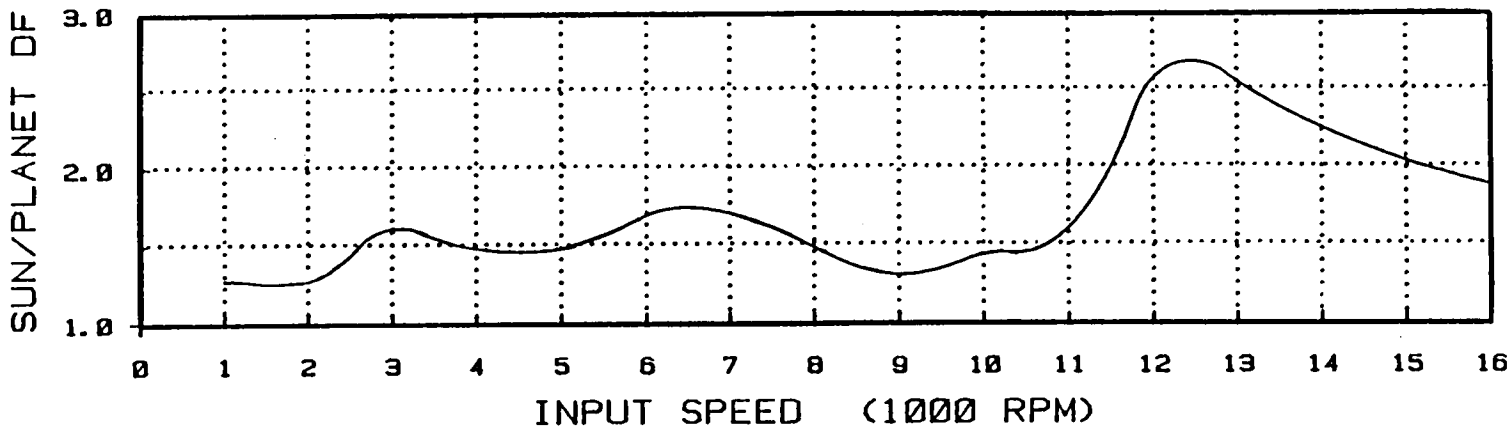


Figure 44 - Dynamic Factors for Floating Sun Gear Configuration with Soft Shafts



(Note: DF > 2.5 denotes unacceptable operating condition)

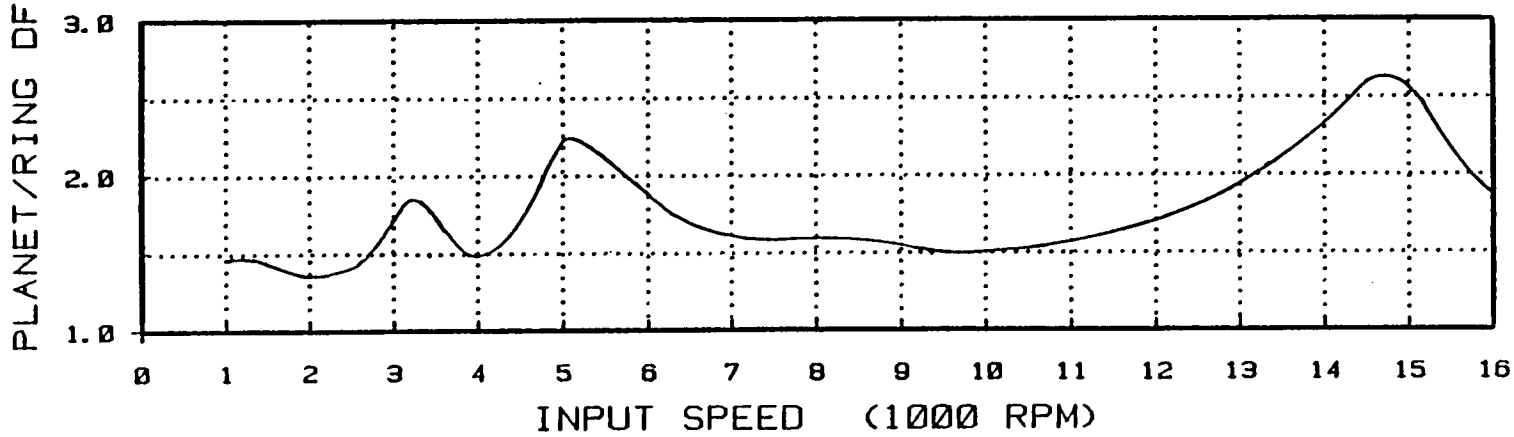


Figure 45 - Dynamic Factors for Fixed Sun Gear Configuration with Soft Shafts

Increased static mesh loads caused a decrease in dynamic loads in the non-resonance region (Fig 46 and 47). Increasing the input torque load from 4500 lb.-in. to 5400 lb-in, increased the operational contact ratio of the sun/planet mesh from 1.578 to 1.623; the planet/ring operational contact changed from 1.891 to 2.021. The planet/ring dynamic loading decreased generally for both systems reflecting the more even mesh stiffness and smoother load transition at the planet/ring mesh.

4.2 Summary and Conclusions

A new methodology has been developed for the static and dynamic load analysis of planetary gear trains (PGT). Prior to this investigation, there were no established methods which used a non-linear tooth mesh stiffness in determining component loading, nor considered the effect of the phase relationships of the variable mesh stiffnesses on the dynamic behavior of the gears. With this newly developed methodology, the design and analysis of planetary gear trains is based on an engineering rather than an empirical approach.

The analysis procedure is applicable to either involute or modified involute spur gearing. It uses non-conjugate gear action caused by deflection of gear teeth and by inherent gear manufacturing and assembly errors. Both fixed and floating sun gear PGT's can be evaluated. The methodology is capable of considering fluctuating output torque, elastic behavior in the entire system as well as in the

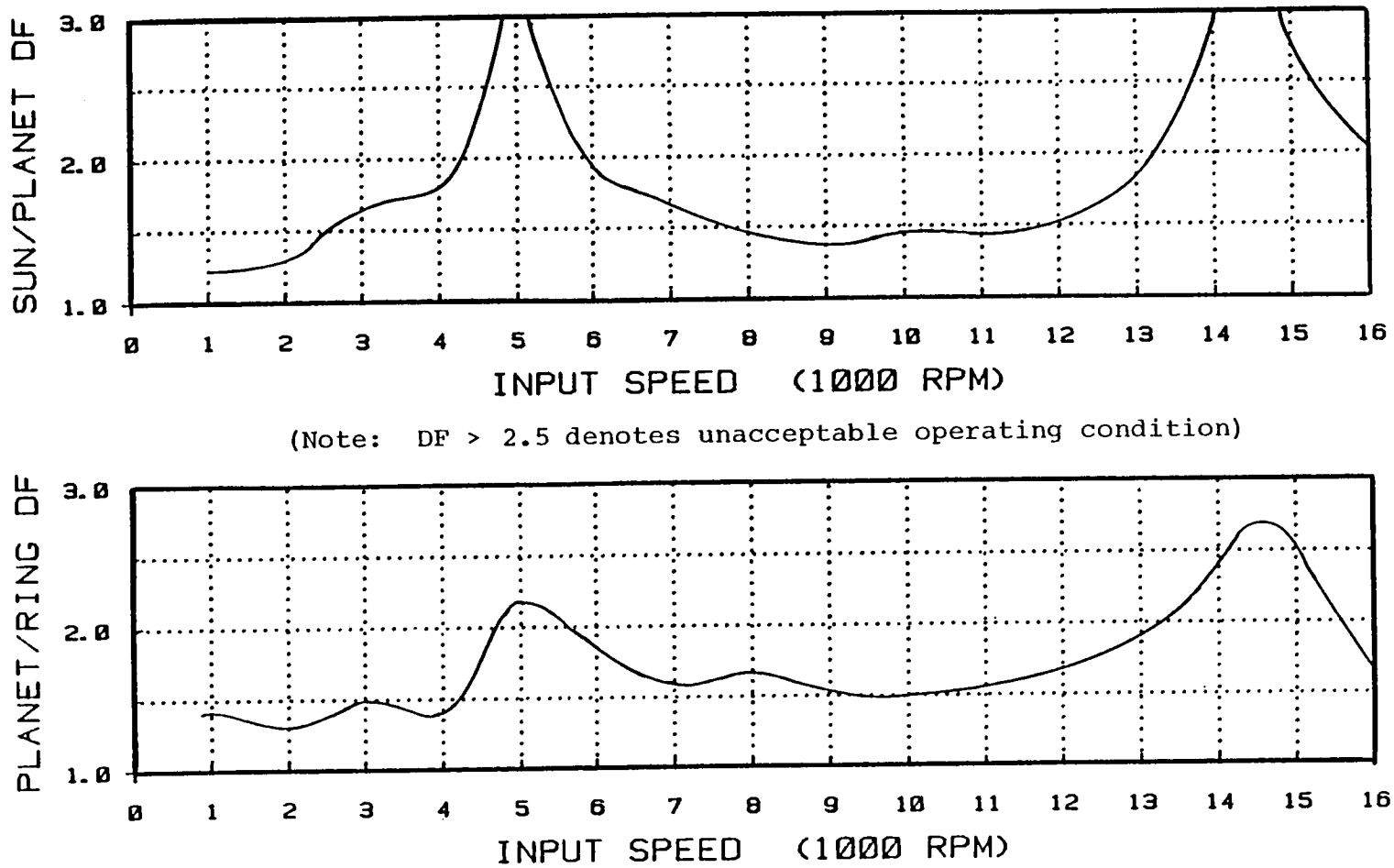


Figure 46 - Dynamic Factors for Floating Sun Gear Configuration with Increased Mesh Loads

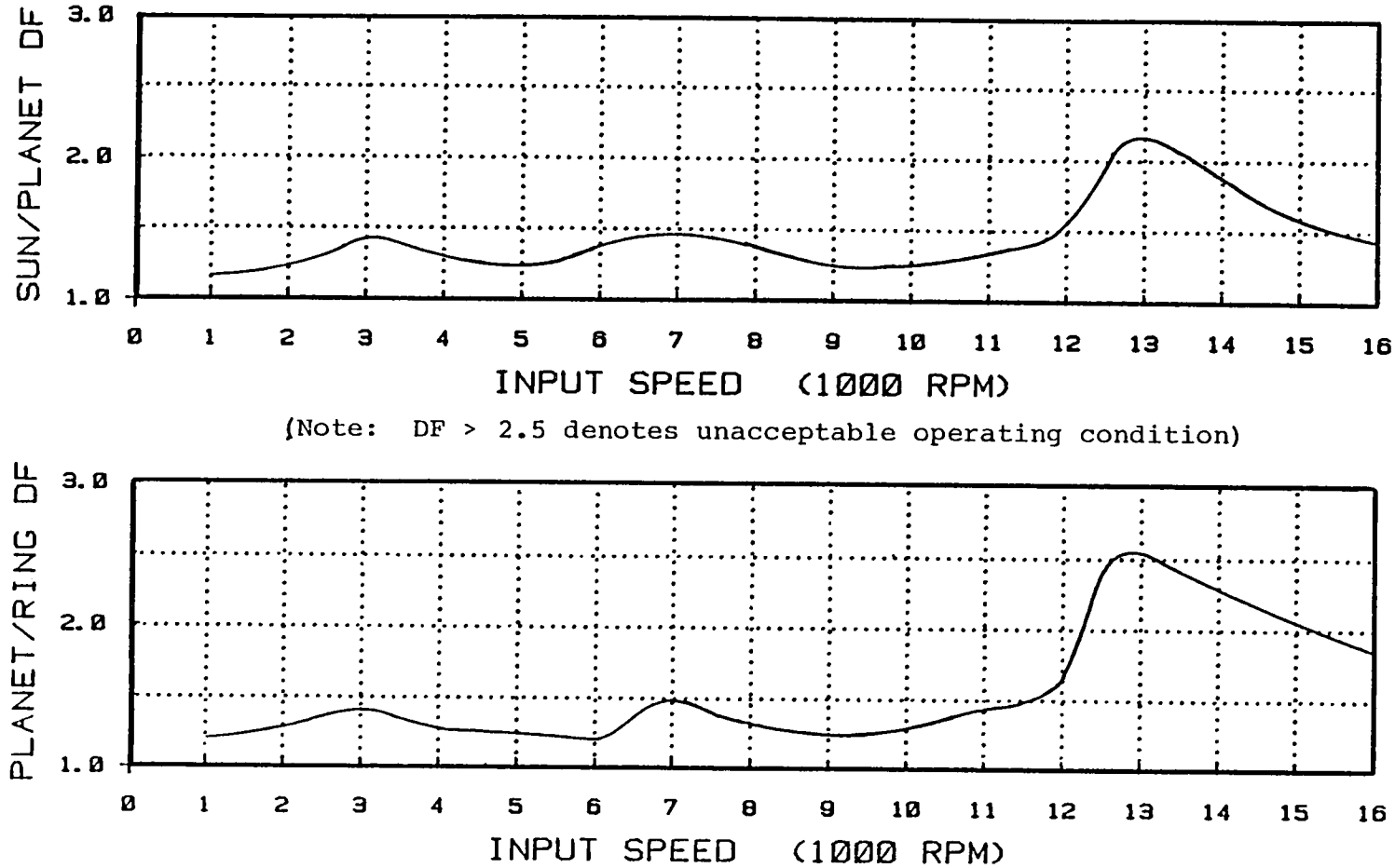


Figure 47 - Dynamic Factors for Fixed Sun Gear Configuration with Increased Mesh Loads

gears, and allows for backlash and possible loss of contact between engaged gear teeth. The above properties are improvements over models used by previous investigators.

The entire methodology has been incorporated into a series of computer programs which comprise a computer-aided-design (CAD) package for the analysis and evaluation of PGT's. Graphical output permits the engineer to quickly and selectively examine all or just portions of the design data base which includes static and dynamic tooth mesh loading, variations in transmission ratio, and sun gear displacements. The results of the parametric computer studies yielded the following conclusions for a typical single stage three-planet planetary gear drive:

a. Under static conditions, a floating sun gear will move to a position such that the mesh loads among the planets are equalized. The locus of the sun center's motion is a determined path with a periodicity equal to one mesh cycle.

b. Under static conditions, a fixed sun gear distributes the load among the planets according to the relative stiffnesses of the effective sun/planet/ring meshes.

c. For pseudo-static conditions, i.e. very low speeds, the floating sun gear can maintain equal load distribution among the planets.

d. At all higher operating speeds, variations and discontinuities in the VVMS, introduce dynamic effects which lead to increased tooth loads. The floating sun gear can no longer maintain equal load distribution among the planets.

e. Allowing the sun gear to float introduces a resonance region where the sun/planet and planet/ring tooth loads are affected by the floating sun gear movement to the point of markedly increased dynamic loads.

f. In non-resonance regions, the floating sun gear configuration generally shows slightly higher dynamic loading than the fixed sun gear configuration, indicating that the floating sun gear motion adds to the dynamic loading in the PGT.

g. At high operating speeds, a resonance region occurs that is indicative of tooth-mesh resonance and results in the highest dynamic loads. This resonance is particularly sensitive to gear tooth errors, and is not affected by the fixity of the sun gear.

h. Altering system components such as the input and output shafts' torsional stiffness, changes the operating characteristics of the PGT. Also, the dynamic loading can be decreased by increasing the system damping and/or the operational contact ratio.

Through the consideration of deformable components, variable operational contact ratio, and time dependent system parameters, the methodology developed during the course of this investigation represents a definite advancement of planetary gear train design technique and assesment. It reflects the latest state-of-the-art understanding of involute and non-involute spur gear performance, and has been adapted for use in a mini-computer based CAD environment. In addition, it establishes the groundwork for further studies on the effect of the dynamic loads on the PGT support structure.

BIBLIOGRAPHY

1. Buckingham,E., "Analytical Mechanics of Gears", Dover Publications, Inc., N.Y., 1949.

2. Tuplin, W.A., "Dynamic Loads on Gear Teeth", Machine Design, p. 203, Oct. 1953.

3. Bollinger,J. G., "Darstellung Des Dynamischen Verhaltens Eines Nichtlinearen Zahnrad - Getriebesystems Auf Dem Analogrechner", Industrie Anzeiger, Nr. 46, S. 961, June 7, 1963

4. Weber,C., "The Deformation of Loaded Gears and the Effect on Their load Carrying Capacity", Sponsored Research (Germany), Report No. 3, 1949.

5. Attia, A. Y. "Deflection of Spur gear Teeth Cut in Their Rims", ASME Paper 63-WA-14, 1963.

6. Kasuba,R. and Evans, J.W. "An Extended Model for Determining Dynamic Loads in Spur Gearing", ASME Paper 80-C2/DET-90, 1980.

7. Pintz, A. "Methodology for Assessing the Dynamic Effects of Internal Spur Gears Drives", DrE Dissertation, Cleveland State University, June, 1982.

8. Lanchester, F.W. and Lanchester, G.H., "Epicyclic Gears", Procedings of the Institute of Mechanical Engineers, Vol. 134, p. 605, 1924

9. Wilson, W.G., "Epicyclic Gearing", Procedings of the Institution of Automobile Engineers, July, 1924, p. 605.

10 Love, P.P., "Epicyclic Gearing", Procedings of the Institution of Mechanical Engineers, Vol. 134, p. 547, 1936.

11. Seager, D.L., "Load Sharing among Planet Gears", SAE Paper 700178, Vol. 3, Sec B, p. 651.

12. Imalle, D. E., "Load Equalization in Planetary Systems", ASME Paper No. 72-PTG-29, 1972.

13. Seager, D.L., "Conditions for the Neutralization of Excitation by the Teeth in Epicyclic Gearing", Journal of Mechanical Engineering Science, Institute of Mechanical Engineering, 1975, pp. 293-298.

14. Cunliffe, F., Smith, J.D., and Welbourne, D.B., "Dynamic Tooth Loads in Epicyclic Gears", ASME Transactions, Journal of Engineering for Industry, May, 1974, pp. 578-584.

15. Chiang, T. and Badgley, R. H. "Reduction of Vibration and Noise Generated by Planetary Ring Gears in Helicopter Aircraft Transmissions", ASME Transactions, Journal of Engineering for Industry, November, 1973, pp. 1149-1158.

16. Tucker, A.I. "The Spring of the Ring", ASME paper 77-125, 1975.

17. Nakahara, T., Takahashi, T., and Masuko, M., "Two Effects of Intermediate Rings on the Load Sharing of Epicyclic Gearing", Paper presented to the International Symposium of Gearing and Power Transmissions, Tokyo, 1981.

18. Botman, M. "Epicyclic Gear Vibrations", ASME Transactions, Journal of Engineering for Industry, August, 1976, pp. 811-815.

19. Hidaka. T., and Terauchi, Y., "Dynamic Behavior of Planetary Gear (1st Report, Load Distribution in Planetary Gear)", Bulletin of the JSME, Vol. 19, No. 138, June 1976, p. 690.

20. Hidaka, T., Terauchi, Y., Nohara, M. and Oshita, J. "Dynamic Behavior of Planetary Gear, 2nd Report, Displacement of Sun Gear and Ring gear", Bulletin of the JSME, Vol. 19, No. 138, December, 1976, pp. 1563-1570.

21. Hidaka, T., Terauchi, Y., Nohara, M. and Oshits, J. "Dynamic Behavior of Planetary Gears, 3rd Report, Displacement of sun Gear and Ring Gear", Bulletin of the JSME, Vol. 19, No. 138, December, 1976, pp. 1563-1570.

22. Hidaka, T., Terauchi, Y., and Nagamura. K. "Dynamic Behavior of Planetary Gears, 7th Report, Influence of the Thickness of the Ring Gear", Bulletin of the JSME, Vol. 22, No. 141, February, 1977, pp. 1142-1149.

23. Hidaka, T., Terauchi, Y. and Fujii, M. "Analysis of dynamic Tooth Load on Planetary Gear Bulletin of JSME, Vol. 23, No. 176, February, 1980, p. 315.

24. Järchow, F. and Vonderschmidt, R. "Tooth-Forces in Planetary Gears", Proceedings of the International Symposium on Gearing and Power Transmissions, 1981, Vol. 11, pp. 327-332.

25. Tordion, G. V., and Gauvin, R., "Dynamic Stability of Two-Stage Gear Train Under the Influence of Variable Meshing Stiffnessess", August 1977, p. 785.

26. Hahn, F. W., "Study of Instantaneous Load to Which Gear Teeth are Subjected", Ph.D. Thesis, Mechanical Engineering Department, University of Illinois, Urbana, Ill., 1969.

27. Radzimovsky, E. I., "Planetary Gear Drives", Machine Design, February 9, 1956, p. 101.

28. Kasuba, R., Evans, J. E., August, R., and Frater, J. L., "Synthesis and Optimization of Spur Gear Tooth Profiles for High Load Capacity", NAS3-18547, NASA Lewis Research Center, Cleveland, Ohio, CR 165163, October 1981.

APPENDIX A

DEVELOPMENT OF THE GEAR VARIABLE-VARIABLE MESH STIFFNESS

This appendix describes the methodology used for the development of the Variable-Variable Mesh Stiffness (VVMS) for the sun/planet and planet/ring gear engagements. The VVMS realistically reflects the gear mesh conditions both as the number of teeth in contact vary and also as the position of contact on the engaged gear teeth vary. An iterative procedure was used to calculate the VVMS as a function of transmitted load, gear tooth profile errors, gear tooth deflections and gear hub torsional deformation, and position of contacting points.

Before the VVMS can be determined, the contacting profiles are digitized into 100 coordinates. Straight line segments connecting the digitized points, which include profile errors and modifications, describe the gear tooth. Deviations from the theoretical involute profile may occur because of manufacturing quality or by design. Also the gear profile may change through extended use, often drastically through surface faults such as pitting or spalling. In terms of an involute profile chart, the involute variation can be expressed as:

$$M = PV(RA), \tag{A1}$$

where M = Deviation from the line of action.

RA = Roll angle measured from the base circle.

PV = Profile variation as a function of RA.

A true involute profile is defined by:

$$M = PV(RA) = 0.$$

The digitized profile points are then used in the iteration process to establish contact points on the gear teeth and also to incorporate appropriate deformations to simulate non-involute action.

A three step procedure is used to determine the VVMS directly. First angular positions are determined for the point of initial contact and point of final disengagement along the theoretical line of action. The contact arc is divided into 49 arc segments, and the two digitized tooth profiles are placed at their respective initial contact positions. This gear pair (GP3), along with the preceding two pairs (GP1,GP2) and the following two pairs (GP4,GP5), is tracked as it rotates through the contact zone. By tracking five gear pairs, it is possible to observe both the mesh behavior composed of all the teeth in contact at a particular time and each individual contacting gear tooth pair, for gear pairs with a contact ratio up to 3.0.

Next, gear tooth deflections are calculated based on candidate contact points established by tracking the movement of GP3 through the mesh arc. Candidate contact points for an individual gear pair are determined by a search procedure

that examines the horizontal distance between 20 to 40 points within an estimated contact zone at every one of the 50 angular positions. Contact occurs if two points are found to be within a prescribed distance of each other. If no profile points meet this criteria, then a no-contact condition is declared for that gear pair at the i^{th} angular position.

The contact gear tooth pair deflection $(k)_i$ can be expressed as:

$$\delta(k)_i = \delta_1(k)_i + \delta_2(k)_i + \delta_H(k)_i \qquad \text{(A2)}$$

where $\delta_1(k)_i$ = Deflection of the k^{th} tooth of gear 1 at mesh arc position i.

$\delta_2(k)_i$ = Deflection of the k^{th} tooth of gear 2 at mesh arc position i.

$\delta_H(k)_i$ = Localized Hertz deformation at the point of contact.

The gear tooth deflection $\delta_1(k)_i$ and $\delta_2(k)_i$ are defined by;

$$\delta_j(k)_i = \delta_{Mj}(k)_i + \delta_{Nj}(k)_i + \delta_{Sj}(k)_i + \delta_{Bj}(k)_i + \delta_{Rj}(k)_i \qquad j=1,2 \qquad \text{(A3)}$$

where δ_M = Gear tooth deflection due to bending.

δ_N = Gear tooth deflection due to normal force.

δ_S = Gear tooth deflection due to shear force.

δ_B = Gear tooth deflection due to deformation of surrounding hub area.

δ_R = Gear tooth deflection due to gross torsion of the rim or hub.

These deflections are considered as equivalent positive profile errors which cause premature engagement and delayed

engagement.

In the third step, the $\delta_1(k)_i$, $\delta_2(k)_i$ and apportioned $\delta_H(k)_i$ deflections areadded to the respective digitized profiles in order to simulate the above gear behavior. The gear pair GP3 is again tracked through the contact arc but now under "loaded and deflected" conditions. The search procedure is repeated during this step. New candidate contact points and positions for five gear pairs are now determined under full load. As a result of the deflections, the contact arc, and therefore the contact ratio of the gear is increased.

For any mesh arc position i, the calculated $k^{\underline{th}}$ gear tooth pair stiffness, $KP(k)_i$, mesh stiffness KG_i, and load sharing incorporates manufactured profile errors, profile modifications, and deflections by means of the iterated numerical solutions.

The individual gear tooth pair stiffness can be expressed as :

$$KP(k)_i = Q(k)_i / \delta(k)_i . \qquad (A4)$$

If the effective errors prevent contact, $KP(k)_i = 0$.

The sum of gear tooth pair stiffnesses for all pairs in contact at position i represents the variable-variable mesh stiffness KG,

$$KG_i = \sum KP(k)_i . \qquad (A5)$$

The load carried by each of the pairs moving through the mesh

arc in the static mode can be determined as:

$$Q(k)_i = \frac{KP(k)_i}{KG_i} Q_t \tag{A6}$$

where Q_t is the total normal static load carried by the gear at any mesh position i in the static mode,

$$Q_t = \sum Q(k)_i . \tag{A7}$$

Determination of the deflected contact points allow the calculation of instantaneous pressure angles and transmission ratios due to non-conjugate gear action. From Figure A-1, the instantaneous parameters for the contact point A' defined by the coordinates UCP(k) and VCP(k) are:

$$PPD' = \text{Distance to instantaneous pitch point} \tag{A8}$$

$$= \frac{RBC1}{\cos \alpha_{A1} + \alpha_{B1}}$$

$$\alpha_{A1} = \arcsin \frac{UCP(k)}{RCP1} \tag{A9}$$

$$\alpha_{A2} = \arcsin \frac{UCP(k)}{RCP2} \tag{A10}$$

$$\alpha_{B1} = \arctan \frac{RCCP1}{RBC1} \tag{A11}$$

$$\alpha' = \text{Instantaneous pressure angle} \tag{A12}$$

$$= \alpha_{A1} + \alpha_{B1}$$

$$\eta = \text{Theoretical (involute) transmission ratio} \tag{A13}$$

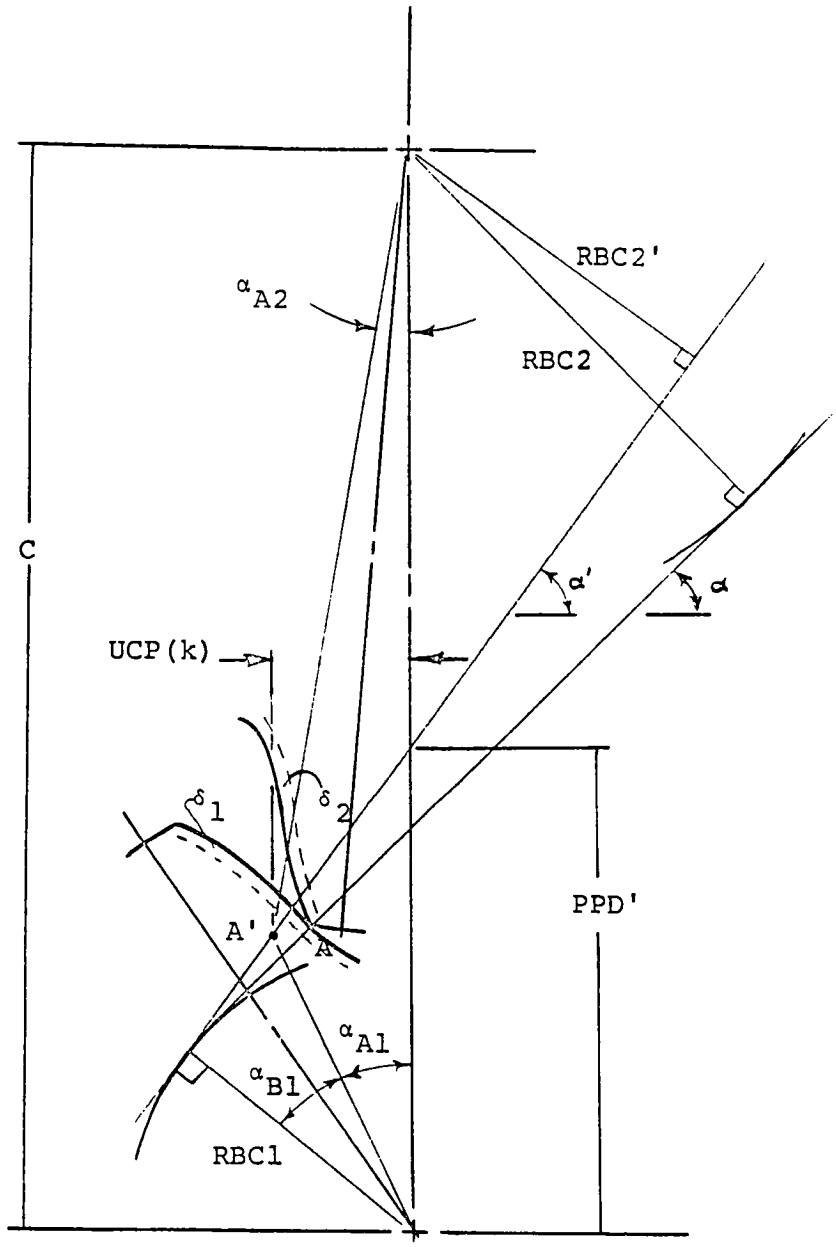


Figure A-1 - Instantaneous Contact Point

$$= \frac{RPC2}{RPC1}$$

$$\eta' = \text{Instantaneous transmission ratio} \qquad \text{(A14)}$$

$$= \frac{(C - PPD')}{PPD'}$$

A similar procedure is used for determining the analogous instantaneous parameters in the external/internal mesh engagements. Where errors interrupt contact, the values of the instantaneous parameters are set equal to the theoretical values. Values for the position dependent mesh stiffness, KGP_i, tooth pair stiffness $KG(k)_i$, and instantaneous transmission ratio, η'_i, are written to data files and used in the subsequent dynamic calculations.

APPENDIX B

EVALUATION OF OTHER TYPES OF EPICYCLIC GEARING

As mentioned in Section 3.4.1, the concept of equivalent gear drives may be extended to the other two types of epicyclic arrangements, star and solar, so that the techniques developed in this investigation may be used to evaluate those drives. This appendix is included to explain how this may be accomplished.

The star drive (Fig. B-1) has a fixed carrier, so no relative motion is introduced in the gear train. The pitch line velocities are equal to the velocities of engagement. Therefore, the absolute angular velocity of the input gear is equal to the actual given input angular velocity. Note that the star drive is a conventional gear train arrangement, meaning no equivalent input velocities must be calculated.

The solar drive (Fig. B-2) has a fixed sun gear, input at the ring gear, and output at the carrier. From Table B-1, the speed ratio of this arrangement is:

$$m_p = \frac{\omega_R}{\omega_C}$$

$$= \frac{N_S}{N_R} + 1.$$

To eliminate the relative angular velocities caused by carrier rotation, the carrier is "stopped" by adding a

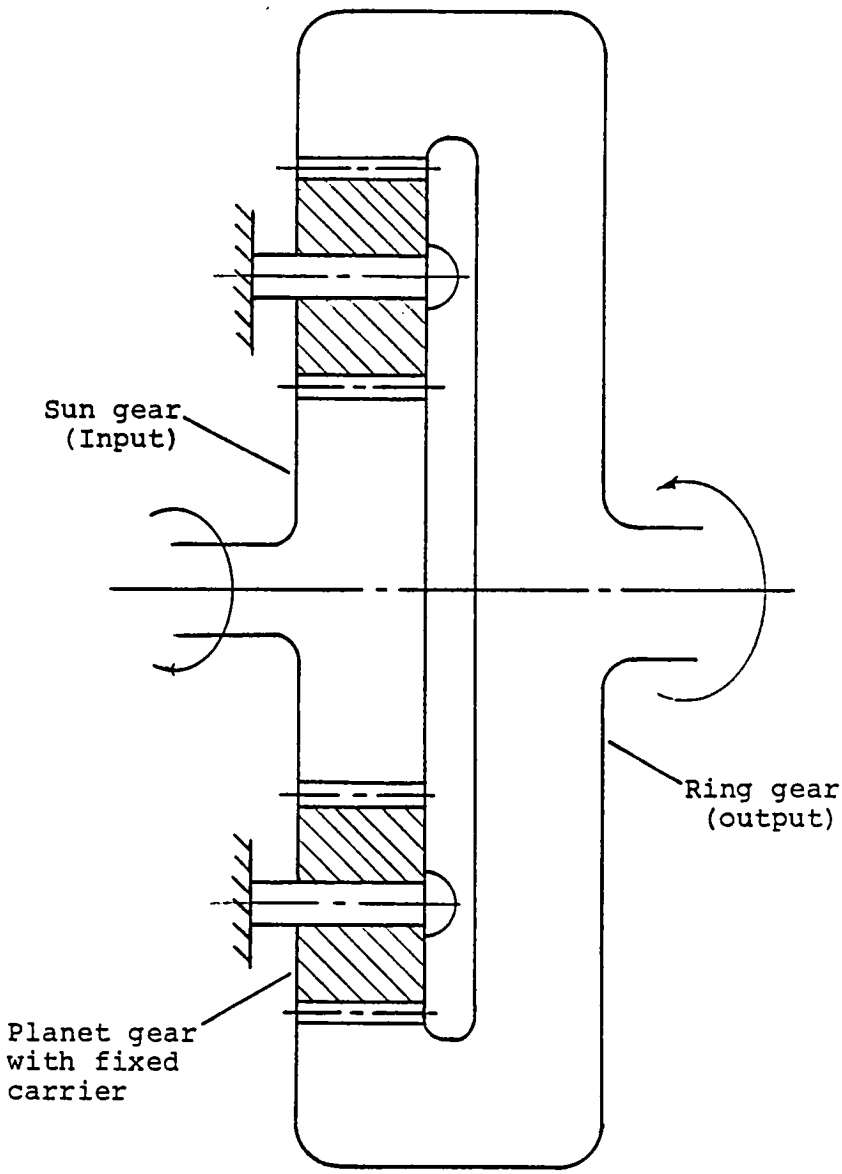


Figure B-1 - Star Arrangement

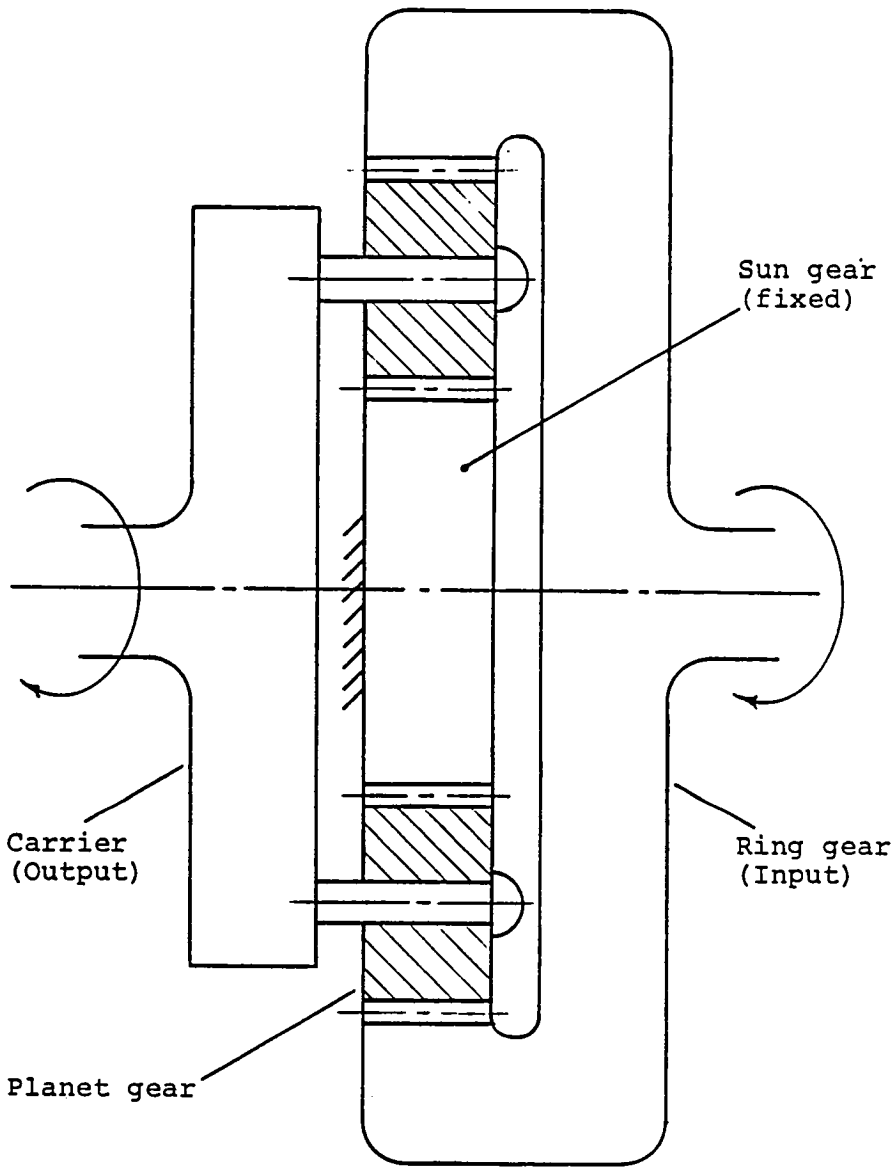


Figure B-2 - Solar Arrangement

Arrangement	Fixed Member	Input Member	Output Member	Over-all Ratio
Planetary	Ring	Sun	Carrier	$\frac{N_R}{N_S} + 1$
Star	Carrier	Sun	Ring	$\frac{N_R}{N_S}$
Solar	Sun	Ring	Carrier	$\frac{N_S}{N_R} + 1$

N_S = Number of sun teeth N_R = Number of ring teeth

Table B-1 - Epicyclic Gear Data

velocity of $-\omega_C$ to all the components. The absolute angular velocity of the input ring gear is now:

$$\omega_R' = \omega_R - \omega_C \tag{B1}$$

$$= \omega_R\left(1 - \frac{1}{m_p}\right)$$

$$= -\frac{N_S}{N_R}\omega_R.$$

The absolute angular velocity of the output gear is:

$$\omega_S' = -\frac{N_R}{N_S}\omega_R' \tag{B2}$$

$$= \omega_R$$

The solar drive arrangement can now be treated as a conventional gear drive with the equivalent sun gear and ring gear velocity equal to ω_R and $-\omega_R N_S/N_R$, respectively.

APPENDIX C

CALCULATION OF TOTAL INTEGRATION TIME AND STEP SIZE

To minimize computer time and yet guarantee stable and accurate numerical solutions, it was decided to base the total integration time and step size on the system's longest and shortest natural periods, respectively. To isolate just the natural frequencies of the gear train, the effect of driver and load ineritas were not considered, although the torsional stiffness values of the input and output shaft were used. The time dependent mesh stiffnesses were assumed to be constant; their value being the average between one and two pairs of teeth in contact. A generalized Jacobi Method was then used to solve the eigenvalue problem:

$$K\phi = \Lambda M\phi$$

where K = Stiffness matrix based on differential equations of motion (see Section 3.5)

ϕ = matrix of eigenvectors

Λ = diagonal matrix of eigenvalues

M = inertia matrix

$$= \begin{bmatrix} J_{sun} & & & & & & -0- \\ & M_{sun} & & & & & \\ & & M_{sun} & & & & \\ & & & J_{p1} & & & \\ & & & & J_{p2} & & \\ & & & & & J_{p3} & \\ -0- & & & & & & J_{ring} \end{bmatrix}$$

```
C     <830613.1126>
FTN4,L
      PROGRAM PGT
C
C******THIS PROGRAM CALCULATES THE NATURAL FREQUENCIES UP TO A 18 D.O.F.
C******MODEL. DEGRESS OF FREEDOM CAN BE ELIMINATED BY SELECTING VERY HIGH
C******STIFFNESS.
C
      IMPLICIT DOUBLE PRECISION (A-H,K,O-Y)
      COMMON /AAA/ A(18,18),B(18,18),X(18,18),EIGV(18),D(18)
      DIMENSION KSP(3),KPR(3),CB(3),SB(3),CA(3),SA(3)
      DIMENSION CG(3),SG(3),ZETA(2,18),ETA(2,18)
      DIMENSION ZX(3,16),ZY(3,16),IG(192),ITITLE(20)
C
C  SIZE, MASS, AND STIFFNESS DATA FOR PLANETARY GEAR SYSTEM
C     ***** GEAR RADII (PITCH CIRCLE) *****
      DATA          RPCS/2.500000D0/, RPCP/2.4000000D0/
C     ***** INERTIA ELEMENTS *****
      DATA          SM/.015000D0/, SJ/.04D0/, PM/.014D0/, PJ/.033D0/
      DATA          CM/9.00D5/, CJ/9.00D5/, RM/1.D-1/, RJ/2.00D0/
C     ***** SUPPORT STIFFNESS ELEMENTS *****
      DATA        KS/1.D0/, KTS/8.85D5/, KP/9.D5/, KTP/0.D0/
      DATA        KC/9.0D5/, KTC/9.0000D5/, KR/9.D5/, KTR/8.85D5/
C     ***** GEAR MESH STIFFNESS *****
      DATA   KSP/3*1915435.D0/, KPR/3*2586154.D0/
      DATA  N/18/, PHI/22.5D0/, PI/3.1415926535897 9D0/
C
      PHI = PI * PHI/180.
   30 WRITE(1,31) RPCS,RPCP
   31 FORMAT(///'RPCS,RPCP',T25,2F13.7)
      WRITE(1,32)
   32 FORMAT('CHANGE?   [Y/N]  _')
      READ(1,21) IQ
      IF(IQ.EQ.1HN) GO TO 40
      WRITE(1,33)
   33 FORMAT('ENTER NEW VALUES')
      READ(1,*) RPCS,RPCP
   40 WRITE(1,3)
      WRITE(1,4)  SM,SJ,PM,PJ,CM,CJ,RM,RJ
      WRITE(1,32)
      READ(1,21) IQ
      IF (IQ.NE.1HY)  GO TO 50
      WRITE(1,33)
      READ(1,*)  SM,SJ,PM,PJ,CM,CJ,RM,RJ
   50 WRITE(1,6)
      WRITE(1,4) KS,KTS,KP,KTP,KC,KTC,KR,KTR
      WRITE(1,32)
      READ(1,21) IQ
      IF (IQ.NE.1HY)  GO TO 60
      WRITE(1,33)
      READ(1,*) KS,KTS,KP,KTP,KC,KTC,KR,KTR
60    WRITE(1,7) KSP,KPR
      WRITE(1,32)
      READ(1,21) IQ
      IF (IQ.NE.1HY)  GO TO 70
```

```
      WRITE(1,33)
      READ(1,*)  KSP,KPR
C
 70   RC = RPCS + RPCP
      RPCR = RPCS + 2*RPCP
      RS = RPCS * DCOS(PHI)
      RP = RPCP * DCOS(PHI)
      RR = RPCR * DCOS(PHI)
      IOUT = 1
C
C     STIFFNESS MATRIX
C
      DO 75 I=1,N
      DO 75 J=1,N
      A(I,J)=0.0
   75 CONTINUE
      A(1,1)   = KTS
      A(2,2)   = KS
      A(3,3)   = KS
      A(13,13) = KTC +3.*KP*RC**2
      A(14,14) = 3.*KP + KC
      A(15,15) = 3.*KP + KC
      A(16,16) = KTR
      A(17,17) = KR
      A(18,18) = KR
C
      DO 100 J=1,3
        BETA = 2./3.*PI*(J-1)
        ALPHA = BETA + PI/2. - PHI
        GAMMA = BETA + PI/2. + PHI
        CB(J) = DCOS(BETA)
        SB(J) = DSIN(BETA)
        CA(J) = DCOS(ALPHA)
        SA(J) = DSIN(ALPHA)
        CG(J) = DCOS(GAMMA)
        SG(J) = DSIN(GAMMA)
C
        I = 1 + 3*J
        A(1,1)    = A(1,1) + KSP(J)*RS**2
        A(1,2)    = A(1,2) + KSP(J)*RS*CA(J)
        A(1,3)    = A(1,3) + KSP(J)*RS*SA(J)
        A(2,2)    = A(2,2) + KSP(J)*CA(J)**2
        A(2,3)    = A(2,3) + KSP(J)*CA(J)*SA(J)
        A(3,3)    = A(3,3) + KSP(J)*SA(J)**2
        A(1,I)    = -KSP(J)*RS*RP
        A(1,I+1)  = -KSP(J)*RS*CA(J)
        A(1,I+2)  = -KSP(J)*RS*SA(J)
        A(2,I)    = -KSP(J)*RP*CA(J)
        A(2,I+1)  = -KSP(J)*CA(J)**2
        A(2,I+2)  = -KSP(J)*CA(J)*SA(J)
        A(3,I)    = -KSP(J)*RP*SA(J)
        A(3,I+1)  = -KSP(J)*CA(J)*SA(J)
        A(3,I+2)  = -KSP(J)*SA(J)**2
        A(I,I)    = KTP + (KSP(J) + KPR(J))*RP**2
        A(I,I+1)  = (KSP(J)*CA(J) - KPR(J)*CG(J))*RF
```

```
        A(I,I+2)   = (KSP(J)*SA(J) - KPR(J)*SG(J))*RP
        A(I+1,I+1)= KP + KSP(J)*CA(J)**2 + KPR(J)*CG(J)**2
        A(I+1,I+2)= KSP(J)*CA(J)*SA(J) + KPR(J)*CG(J)*SG(J)
        A(I+2,I+2)= KP + KSP(J)*SA(J)**2 + KPR(J)*SG(J)**2
        A(13,14)   = A(13,14) - KP*RC*SB(J)
        A(13,15)   = A(13,15) + KP*RC*CB(J)
        A(I+1,13) = KP*RC*SB(J)
        A(I+1,14) = -KP
        A(I+2,13) = -KP*RC*CB(J)
        A(I+2,15) = -KP
        A(16,16)   = A(16,16) + KPR(J)*RR**2
        A(16,17)   = A(16,17) - KPR(J)*RR*CG(J)
        A(16,18)   = A(16,18) - KPR(J)*RR*SG(J)
        A(17,17)   = A(17,17) + KPR(J)*CG(J)**2
        A(17,18)   = A(17,18) + KPR(J)*CG(J)*SG(J)
        A(18,18)   = A(18,18) + KPR(J)*SG(J)**2
        A(I,16)    = -KPR(J)*RP*RR
        A(I,17)    =  KPR(J)*RP*CG(J)
        A(I,18)    =  KPR(J)*RP*SG(J)
        A(I+1,16) =  KPR(J)*RR*CG(J)
        A(I+1,17) = -KPR(J)*CG(J)**2
        A(I+1,18) = -KPR(J)*CG(J)*SG(J)
        A(I+2,16) =  KPR(J)*RR*SG(J)
        A(I+2,17) = -KPR(J)*CG(J)*SG(J)
        A(I+2,18) = -KPR(J)*SG(J)**2
C
C       INERTIA ELEMENTS
C
        IF (J.GE.2)    B(J,J) = SM
        B(I,I)     = PJ
        B(I+1,I+1)= PM
        B(I+2,I+2)= PM
        IF (J.GE.2)    B(J+12,J+12) = CM
        IF (J.GE.2)    B(J+15,J+15) = RM
  100   CONTINUE
        B(1,1)    = SJ
        B(13,13) = CJ
        B(16,16) = RJ
C
C       FILL SYMMETRIC MATRIX
C
        DO 200 I=1,N
          DO 200 J=1,N
            A(J,I) = A(I,J)
  200   CONTINUE
        IF (IOUT.NE.6)  IOUT = 1
  401   CALL JACOB(N,1.0D-12,20,0,IOUT)

C       PRINT RESULTS
C
  601   WRITE(1,2)
        READ(1,*) IOUT
        IF(IOUT.NE.1 .AND. IOUT.NE.6)  GO TO 601
C
        WRITE(IOUT,31) RPCS,RPCP
```

```
      WRITE(IOUT,3)
      WRITE(IOUT,4)  SM,SJ,PM,PJ,CM,CJ,RM,RJ
      WRITE(IOUT,6)
      WRITE(IOUT,4) KS,KTS,KP,KTP,KC,KTC,KR,KTR
      WRITE(IOUT,7) KSP,KPR
C
      DO 650 J=1,N
      EIGV(J) = DSQRT(EIGV(J))/(2*PI)
      PERIOD  = 1./EIGV(J)
      WRITE(IOUT,649) J,EIGV(J),PERIOD
  649 FORMAT(I5,5X,'FREQUENCY',F12.5,' Hz',5X,'PERIOD',F11.7,' sec')
  650 CONTINUE
      WRITE(1,660)
  660 FORMAT('PRINT MODE SHAPES?  [Y/N] _')
      READ(1,21) IPICK
      IF (IPICK.EQ.1HN) GO TO 801
C
       WRITE(IOUT,11) (J, J=1,N)
       WRITE(IOUT,16)   (EIGV(J), J=1,N)
       WRITE(IOUT,12)
C
C  CALCULATING LOCAL COORDINATE DISPLACEMENTS FOR PLANETS #2 & #3
C  (FOR PLANET #1, LOCAL & GLOBAL COORDINATES ARE IDENTICAL.)
C
       DO 800 I=1,N
         DO 700 J=2,3
           JJ = 3*J +2
           ZETA(J-1,I) =  X(JJ,I)*CB(J) + X(JJ+1,I)*SB(J)
           ETA(J-1,I)  = -X(JJ,I)*SB(J) + X(JJ+1,I)*CB(J)
  700    CONTINUE
       WRITE(IOUT,15) I, (X(I,J), J=1,N)
       IF (I.EQ.8 .OR. I.EQ.11) WRITE(IOUT,16)  (ZETA(I/5,J),J=1,N)
       IF (I.EQ.9 .OR. I.EQ.12) WRITE(IOUT,16)  (ETA(I/6,J),J=1,N)
  800  CONTINUE
C
  801 WRITE(1,802)
  802 FORMAT(///'AGAIN?   [Y/N]  _')
      READ(1,21) IPICK
      IF (IPICK.NE.1HN) GO TO 30
C
      CONTINUE
    1 FORMAT(// 'MODE NO.', I5, / (/3(2F12.4,6X)))
    2 FORMAT(/'LU FOR OUTPUT DEVICE?  [1-SCR  6-PRT]  _'//)
    3 FORMAT(/'INERTIA ELEMENTS:  M-(MASS) & J-(MOMENT)')
    4 FORMAT('SUN',T25,2F13.5/'PLANET',T25,2F13.5/'CARRIER',T25,2F13.5/
     &       'RING',T25,2F13.5)
    6 FORMAT(/'SUPPORT STIFFNESSES: K-(TRANSLATION) & Kt-(TORQUE)')
    7 FORMAT(/'MESH STIFFNESSES'/'KSP  (SUN/PLANET)',T25,3F13.3/
     &        'KPR (PLANET/RING)',T25,3F13.3)
   11 FORMAT(//// 'NATURAL FREQUENCIES (HZ): 18 DOF SYSTEM'// 18F7.1)
   12 FORMAT(// 'EIGENVECTORS OF SYSTEM')
   15 FORMAT(/ I2,18F7.4)
   16 FORMAT(2X,18F7.1)
   21 FORMAT(A1)
   22 FORMAT(/ 'TO CALCULATE MODAL DISPLACEMENT OF ANY EIGENVECTOR,' /
```

```
     & 'ENTER ITS NUMBER (1-18), ENTRY OF '0' TERMINATES PROGRAM  _')
 23   FORMAT(/ 'FOR GRAPHICS OUTPUT ON SCREEN, ENTER '1'' / 'FOR OUTPUT
     &ON 4-PEN PLOTTER, *****FIRST****LOAD 8.5 X 11 PAPER VERTICALLY' /
     & 'AGAINST LEFT & BOTTOM STOPS, THEN ENTER:' / ''2,0' FOR FULL PAGE
     & DRAWING' / ''2,1' FOR UPPER HALF PAGE DRAWING, OR' /
     & ''2,2' FOR LOWER HALF PAGE DRAWING  _')
 26   FORMAT(/ 'ENTER TITLE (MAX. 40 CHARACTERS-TO HERE)')
 27   FORMAT(20A2)
 999  END
C
C
      BLOCK DATA AAA
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      COMMON /AAA/ A(18,18),B(18,18),X(18,18),EIGV(18),D(18)
      DATA  A/324*0.D0/, B/324*0.D0/, X/324*0.D0/, EIGV/18*0.D0/
      END
```

```
C
C
      SUBROUTINE JACOB(N,RTOL,NSMX,IFPR,IOUT)
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      COMMON /AAA/ A(18,18),B(18,18),X(18,18),EIGV(18),D(18)
      DIMENSION ITIME(5),JTIME(5),NTIME(5)
      IF (IOUT.EQ.0)   IOUT = 1
C
C  ***************************************************************
C  *                                                             *
C  *  EIGENVALUE SUBROUTINE (USING GENERALIZED JACOBI ITERATION) *
C  *                                                             *
C  *  INPUT VARIABLES:                                           *
C  *    A(18,18)     = STIFNESS MAT'X (MUST BE POSITIVE DEFINITE)*
C  *    B(18,18)     = MASS MAT'X (MUST BE POSITIVE DEFINITE)    *
C  *    N            = ORDER OF MATRICES  (MUST BE <= 18)        *
C  *    RTOL         = CONVERGENCE TOLERENCE (TYP. 1.0D-12)      *
C  *    NSMX         = LIMITING NUMBER OF SWEEPS (TYP. 15)       *
C  *    IFPR         = FLAG FOR PRINTING INTERMEDIATE VALUES     *
C  *                   (IF .EQ. 0, WILL NOT PRINT)               *
C  *                                                             *
C  *  OUTPUT VARIABLES:                                          *
C  *    A(N,N)       = DIAGONALIZED STIFFNESS MATRIX             *
C  *    B(N,N)       = DIAGONALIZED MASS MATRIX                  *
C  *    X(N,N)       = EIGENVECTORS STORED COLUMNWISE            *
C  *    EIGV(N)      = EIGENVALUES                               *
C  *                                                             *
C  *    D(18)        = WORKING VECTOR                            *
C  *                                                             *
C  ***************************************************************
C
C  INITIALIZE EIGENVALUE & EIGENVECTOR MATRICES
C
      DO 10 I=1,N
        IF (A(I,I).GT.0. .AND. B(I,I).GT.0.) GO TO 4
        WRITE(IOUT,2020)  I,A(I,I),B(I,I)
        STOP 1
 4      D(I) = A(I,I)/B(I,I)
        EIGV(I) = D(I)
 10   CONTINUE
      DO 30 I=1,N
        DO 20 J=1,N
 20     X(I,J) = 0.
 30   X(I,I) = 1.
      IF (N.EQ.1)  RETURN
C
C  INITIALIZE SWEEP COUNTER & BEGIN ITERATION
C
                      IQ = 0
C  (INTERMEDIATE       WRITE(1,2997)
C  VALUE DUMP)         READ(1,*) IQ
      CALL EXEC(11,ITIME)
      NSWEEP = 0
      NR = N-1
 40   NSWEEP = NSWEEP + 1
```

```
      IF (IFPR.EQ.1) WRITE(IOUT,2000) NSWEEP
C
C  CHECK WHETHER PRESENT OFF-DIAGONAL ELEMENT REQUIRES ZEROING.
C
      EPS = (.01**NSWEEP)**2
      DO 210 J=1,NR
        JJ = J+1
        DO 210 K=JJ,N
                IF (IQ.EQ.1) WRITE(IOUT,2998) J,K
          ATOL = (A(J,K)**2)/(A(J,J)*A(K,K))
          BTOL = (B(J,K)**2)/(B(J,J)*B(K,K))
          IF ((ATOL.LT.EPS).AND.(BTOL.LT.EPS)) GO TO 210
C
C  IF ZEROING REQ'D, WE CALCULATE ROTATION MATX. ELEMENTS 'CA' & 'CG'.
C
          AKK = A(K,K)*B(J,K) - B(K,K)*A(J,K)
          AJJ = A(J,J)*B(J,K) - B(J,J)*A(J,K)
          AB  = A(J,J)*B(K,K) - A(K,K)*B(J,J)
          CHK = (AB**2 + 4.*AKK*AJJ)/4.
          IF (CHK) 50,60,60
 50       WRITE(IOUT,2020) K,CHK
          STOP 2
 60       SQCH = DSQRT(CHK)
          D1 = AB/2. + SQCH
          D2 = AB/2. - SQCH
          DEN = D1
          IF (DABS(D2).GT.DABS(D1))  DEN = D2
          IF (DEN) 80,70,80
 70       CA = 0.
          CG = -A(J,K)/A(K,K)
          GO TO 90
 80       CA = AKK/DEN
          CG = -AJJ/DEN
                IF (IQ.EQ.1) WRITE(IOUT,2999) K,CA,CG
C
C  PERFORM GENERALIZED ROTATION TO ZERO PRESENT OFF-DIAGONAL ELEMENT.
C
 90       IF (N-2) 100,190,100
 100      JP1 = J+1
          JM1 = J-1
          KP1 = K+1
          KM1 = K-1
          IF (JM1-1) 130,110,110
 110      DO 120 I=1,JM1
            AJ = A(I,J)
            BJ = B(I,J)
            AK = A(I,K)
            BK = B(I,K)
            A(I,J) = AJ + CG*AK
            B(I,J) = BJ + CG*BK
            A(I,K) = AK + CA*AJ
            B(I,K) = BK + CA*BJ
                    IF (IQ.EQ.1) WRITE(IOUT,2999) I,(A(I,II),II=1,N)
 120      CONTINUE
 130      IF (KP1-N) 140 140,160
```

```
 140      DO 150 I=KP1,N
            AJ = A(J,I)
            BJ = B(J,I)
            AK = A(K,I)
            BK = B(K,I)
            A(J,I) = AJ + CG*AK
            B(J,I) = BJ + CG*BK
            A(K,I) = AK + CA*AJ
            B(K,I) = BK + CA*BJ
             IF (IQ.EQ.1) WRITE(IOUT,2999) I,(A(I,II),II=1,N)
 150      CONTINUE
 160      IF (JP1-KM1) 170,170,190
 170      DO 180 I=JP1,KM1
            AJ = A(J,I)
            BJ = B(J,I)
            AK = A(I,K)
            BK = B(I,K)
            A(J,I) = AJ + CG*AK
            B(J,I) = BJ + CG*BK
            A(I,K) = AK + CA*AJ
            B(I,K) = BK + CA*BJ
             IF (IQ.EQ.1) WRITE(IOUT,2999) I,(A(I,II),II=1,N)
 180      CONTINUE
 190      AK = A(K,K)
          BK = B(K,K)
          A(K,K) = AK + 2.*CA*A(J,K) + A(J,J)*CA**2
          B(K,K) = BK + 2.*CA*B(J,K) + B(J,J)*CA**2
          A(J,J) = A(J,J) + 2.*CG*A(J,K) + AK*CG**2
          B(J,J) = B(J,J) + 2.*CG*B(J,K) + BK*CG**2
          A(J,K) = 0.
          B(J,K) = 0.
C
C  UPDATE EIGENVECTOR MATRIX AFTER EACH ROTATION.
C
               DO 200 I=1,N
                 XJ = X(I,J)
                 XK = X(I,K)
                 X(I,J) = XJ + CG*XK
                 X(I,K) = XK + CA*XJ
              IF (IQ.EQ.1) WRITE(IOUT,2999) I,(X(I,II),II=1,N)
 200            CONTINUE
 210  CONTINUE
C
C  UPDATE EIGENVALUES AFTER EACH SWEEP.
C
      DO 220 I=1,N
        IF (A(I,I).GT.0. .AND. B(I,I).GT.0.) GO TO 220
        WRITE(IOUT,2020)  I,A(I,I),B(I,I)
C       STOP 3
 220  EIGV(I) = A(I,I)/B(I,I)
      IF (IFPR.EQ.0) GO TO 230
      WRITE(IOUT,2030)
      WRITE(IOUT,2010)  (EIGV(I), I=1,N)
C
C  CHECK FOR CONVERGENCE.
```

```
C
 230  DO 240 I=1,N
        TOL = RTOL*D(I)
        DIF = DABS(EIGV(I) - D(I))
        IF (DIF.GT.TOL) GO TO 280
 240  CONTINUE
C
C  CHECK ALL OFF-DIAGONAL ELEMENTS TO DETERMINE IF ANOTHER SWEEP REQ'D.
C
      EPS = RTOL**2
      DO 250 J=1,NR
       JJ = J+1
       DO 250 K=JJ,N
         EPSA = (A(J,K)**2)/(A(J,J)*A(K,K))
         EPSB = (B(J,K)**2)/(B(J,J)*B(K,K))
         IF ((EPSA.LT.EPS).AND.(EPSB.LT.EPS)) GO TO 250
         GO TO 280
 250   CONTINUE
C
C  FILL BOTTOM TRIANGLE OF A & B MATRICES & NORMALIZE EIGENVECTORS.
C
 255  DO 270 I=1,N
        XX = 0.
        DO 260 J=1,N
          XX = XX + X(J,I)**2
          A(J,I) = A(I,J)
          B(J,I) = B(I,J)
 260    CONTINUE
        DO 270 J=1,N
          X(J,I) = X(J,I)/DSQRT(XX)
 270  CONTINUE
      CALL EXEC(11,JTIME)
      DO 285 I=1,5
  285 NTIME(I) = JTIME(I) - ITIME(I)
      WRITE(IOUT,2040) ITIME,JTIME,NTIME,NSWEEP
      RETURN
C
C  UPDATE "D" MATRIX, START NEW SWEEP, IF ALLOWED.
C
 280  DO 290 I=1,N
 290    D(I) = EIGV(I)
      IF (NSWEEP.LT.NSMX) GO TO 40
      GO TO 255
 2000 FORMAT (/ 'SWEEP NUMBER =',I4)
 2010 FORMAT(4G20.12)
 2020 FORMAT(/ '***ERROR***MATRICES NOT POSITIVE ',
     &  'DEFINITE ...VALUE OF I =' I5,' A(I,I) & B(I,I) =',2F12.4)
 2030 FORMAT(/ ''JACOB': CURRENT EIGENVALUES...' /)
 2040 FORMAT(/ 'STARTING TIME  =', 5I5 / 'FINISH TIME    =', 5I5 /
     &  'COMPUTING TIME =', 5I5, '   NO. OF SWEEPS =', I5)
 2997                FORMAT(/ 'IF CALC. DISP. DESIRED, ENTER 1 ELSE 0')
 2998                FORMAT(/ 'TOP OF LOOP: J,K =', 5I8)
 2999                FORMAT( 'TEST', I5,(6F12.3))
      END
```

APPENDIX D

```
C     <830610.1521>
FTN4,L
      PROGRAM EXPRO
C
C     THIS PROGRAM GENERATES GEAR TOOTH PROFILE COORDINATES
C
      COMMON/C1/X1(100),X2(100),Y1(100),Y2(100),THETA1(100),THETA2(100),
     1          W1(100),W2(100),Z1(100),Z2(100),RR1(100),RR2(100)
      COMMON/C2/E1,G1,PR1,YP1,RRO1,RI1,FW1,XMIN1,
     1          E2,G2,PR2,YP2,RRO2,RI2,FW2,XMIN2
      COMMON/C3/PI,PHI,TG1,TG2,TP,PD1,PD2,RPC1,RPC2,RAC1,RAC2,TIN,RBC1,
     1          RBC2,C,SP,EP
C
      DIMENSION LENGTH(2)
      DIMENSION TITLE(10)
      DIMENSION IBUF(40),IDCB1(144),INAM(3),ISIZE(2)
      DIMENSION IDATA(2)
C
      REAL KL,LEI,KEI
C
C*****NO LIBRARY ARCCOS OR ARCSON FUNCTIONS REQUIRES FOLLOWING FUNCTIONS
C
      ARSIN(X)=ATAN(X/SQRT(1.-X**2))
      ARCOS(X)=PI/2.-ATAN(X/SQRT(1.-X**2))
C
C
      DATA LENGTH/2HIN,2HMM/
C
C FOLLOWING DATA STATEMENTS ARE FOR INPUT
C
      DATA DP/5./,DELTP/.01/,PHID/22.5/
      DATA MODCOD/0/
      DATA AD1,AD2,WD1,WD2,GRRF1,GRRF2/2*.200,2*.470000,2*.020000/
C
C     DATA DP,DELTP,PHID/8.4666667,.01,22.5/
C     DATA MODCOD/0/
C     DATA AD1,AD2,WD1,WD2/2*.11811,.269,.265748/
C     DATA GRRF1,GRRF2/2*.05/,RPMIN/1000./
C
      WRITE(1,2)
    2 FORMAT('INPUT LU FOR OUTPUT DEVICE,PER1,CYC1,PAP1')
      READ(1,*) LU,PER1,CYC1,PAP1
C
      TODEGR=180./PI
    6 PHI=PHID*PI/180.
      TOUT=TIN*TG2/TG1
      RPMOUT=RPMIN*TG1/TG2
      G1=0.5*E1/(1.+PR1)
      G2=0.5*E2/(1.+PR2)
      PD1=TG1/DP
      PD2=TG2/DP
      RPC1=0.5*PD1
      RPC2=0.5*PD2
    7 IF(RI1.EQ.0.0) RI1=(16.*TIN/(PI*TAUMAX))**(1./3.)*0.5
      IF(RI2.EQ.0.0) RI2=(16.*TOUT/(PI*TAUMAX))**(1./3.)*0.5
```

```
      IF(JG1.EQ.0.) JG1=.5*GAMA1*PI*FW1*RPC1**4/386.
      IF(JG2.EQ.0.) JG2=.5*GAMA2*PI*FW2*RPC2**4/386.
      C=RPC1+RPC2
      CP=PI/DP
      BP=CP*COS(PHI)
      RF1=.7*(GRRF1+(WD1-AD1-GRRF1)**2/(.5*PD1+WD1-AD1-GRRF1))
      RF2=.7*(GRRF2+(WD2-AD2-GRRF2)**2/(.5*PD2+WD2-AD2-GRRF2))
      RAC1=RPC1+AD1
      RAC2=RPC2+AD2
      RRC1=RAC1-WD1
      RRC2=RAC2-WD2
      RI1 =RRC1
      RI2 =RRC2
      RBC1=RPC1*COS(PHI)
      RBC2=RPC2*COS(PHI)
C
      L1=100
      L2=L1
      P=TIN/RBC1
C
C     NEXT SEGMENT OF CODE IS FROM MOD
C
C     TG-----NUMBER OF TEETH
C     DP-----DIAMETRAL PITCH
C     PHI----PRESSURE  ANGLE
C     AD-----ADDENDUM
C     WD-----WHOLE DEPTH (APPROXIMATE)
C     GRRF---GENERATING RACK EDGE RADIUS
C     1------IDENTIFIES GEAR 1
C     2------IDENTIFIES GEAR 2
C     RF-----FILLET RADIUS
C
C
C*****READ PROFILE MODIFICATIONS (IF ANY)
C
C
C
C*****CHECK FOR INTERFERENCE
C
  199 RCHK1 = SQRT(RBC1**2 + (C*SIN(PHI))**2)
      RCHK2 = SQRT(RBC2**2 + (C*SIN(PHI))**2)
      IF (RAC1.GT.RCHK1.OR.RAC2.GT.RCHK2) INF=2
      IF (INF.EQ.2) GO TO 4563
      X=RF1/(RRC1+RF1)
      ALPHA1=ARSIN(X)
      X=RF2/(RRC2+RF2)
      ALPHA2=ARSIN(X)
      RTF11=(RRC1+RF1)*COS(ALPHA1)
      RTF22=(RRC2+RF2)*COS(ALPHA2)
C
      WRITE(LU,200) TG1,TG2,PHI,DP,RPC1,RPC2,RBC1,RBC2,RRC1,RRC2,RF1,
     &RF2,RI1,RI2,ALPHA1,ALPHA2,RTF11,RTF22
200   FORMAT('TG1,TG2         :',2F14.7/
     &       'PHI,DP          :',2F14.7/
     &       'RPC1,RPC2       :',2F14.7/
```

```
     &       'RBC1,RBC2       :',2F14.7/
     &       'RRC1,RRC2       :',2F14.7/
     &       'RF1,RF2         :',2F14.7/
     &       'RI1,RI2         :',2F14.7/
     &       'ALPHA1,ALPHA2   :',2F14.7/
     &       'RTF11,RTF22     :',2F14.7/)
C
C     CALCULATION OF LIMIT RADII (RLM1 AND RLM2)
C
      AUX1=ARCOS(RBC1/RAC1)
      AUX2=ARCOS(RBC2/RAC2)
      CI1=RAC2*SIN(AUX2)-RPC2*SIN(PHI)
      CI2=RAC1*SIN(AUX1)-RPC1*SIN(PHI)
      RALR1=ATAN((RPC1*SIN(PHI)-CI1)/RBC1)
      RALR2=ATAN((RPC2*SIN(PHI)-CI2)/RBC2)
      RLM1=RBC1/COS(RALR1)
      RLM2=RBC2/COS(RALR2)
C
      WRITE(LU,15) AUX1,AUX2,CI1,CI2,RALR1,RALR2,RLM1,RLM2
15    FORMAT('AUX1,AUX2      :',2F14.7/
     1       'CI1,CI2         :',2F14.7/
     2       'RALR1,RALR2     :',2F14.7/
     3       'RLM1,RLM2       :',2F14.7/)
      RR1=RAC1-RLM1
      RR2=RAC2-RLM2
      IF (RLM1.LE.RRC1) RR1=RAC1-RTF11
      IF (RLM2.LE.RRC2) RR2=RAC2-RTF22
  220 FORMAT('0',2X,'NOTE:  RADIUS OF THEORETICAL LAST POINT OF CONTACT
     &ON GEAR 1 IS LESS THAN THE ROOT CIRCLE RADIUS.'/
     &' TO AVOID INTERFERENCE PROBLEMS, THIS TOOTH SHOULD BE UNDERCUT'/)
  221 FORMAT('0',2X,'NOTE:  RADIUS OF THEORETICAL LAST POINT OF CONTACT
     &ON GEAR 2 IS LESS THAN THE ROOT CIRCLE RADIUS.'/
     &' TO AVOID INTERFERENCE PROBLEMS, THIS TOOTH SHOULD BE UNDERCUT'/)
C
      LI1=IFIX((RR1/(RAC1-RRC1))*L1)
      LI2=IFIX((RR2/(RAC2-RRC2))*L2)
      RINCI1=RR1/FLOAT(LI1-1)
      RINCI2=RR2/FLOAT(LI2-1)
C
C
C     RA1-----ROLL ANGLE,GEAR 1
C     RATM1---LENGTH OF TIP MODIFICATION IN DEGREES OF ROLL,GEAR 1
C     RAT1----ROLL ANGLE AT TIP OF GEAR 1
C     RATI1---ROLL ANGLE AT TOP OF INVOLUTE,GEAR 1
C     RABI1----ROLL ANGLE AT THE BOTTOM OF INVOLUTE,GEAR 1
C     RABM1---LENGTH OF ROOT MODIFICATION IN DEGREES OF ROLL,GEAR 1
C     PATM1---MAGNITUDE OF PARABOLIC MODIFICATION AT THE TIP,GEAR 1
C     PABM1---MAGNITUDE OF PARABOLIC MODIFICATION AT THE BOTTOM,GEAR 1
C     STTM1---MAGNITUDE OF STRAIGHT LINE MODIFICATION AT THE TIP,GEAR 1
C     STBM1---MAGNITUDE OF STRAIGHT LINE MODIFICATION AT THE BOTTOM,GEAR
C     PER1---MAX MANUFACTURED PROFILE ERROR,GEAR 1
C     PAP1---ANGLE FROM START OF SIN. ERROR TO START OF INVOLUTE,GEAR 1
C     RTI1----RADIUS TO TOP OF INVOLUTE,GEAR 1
C     RBI1----RADIUS TO BOTTOM OF INVOLUTE,GEAR 1
C
```

```
C
C*****CALCULATION OF ROLL ANGLES TO INVOLUTE TOP, PITCH, AND BOTTOM; AND
C*****RADIAL DISTANCES TO (UN)MODIFIED INVOLUTE TOP, PITCH, AND BOTTOM
C
      RAT1=TODEGR*SQRT((RAC1/RBC1)**2 - 1.)
      RAT2=TODEGR*SQRT((RAC2/RBC2)**2 - 1.)
      RAM1=RAT1-RATM1
      RAM2=RAT2-RATM2
      RTI1=RBC1*SQRT((RAM1/TODEGR)**2 + 1.)
      RTI2=RBC2*SQRT((RAM2/TODEGR)**2 + 1.)
C
      RAP=TODEGR*TAN(PHI)
      RATIP1=RAM1-RAP
      RATIP2=RAM2-RAP
C
      RABI1=TODEGR*SQRT((RLM1/RBC1)**2 - 1.)
      RABI2=TODEGR*SQRT((RLM2/RBC2)**2 - 1.)
      RAN1=RABI1+RABM1
      RAN2=RABI2+RABM2
      RBI1=RBC1*SQRT((RAN1/TODEGR)**2 + 1.)
      RBI2=RBC2*SQRT((RAN2/TODEGR)**2 + 1.)
C
C*****CALCULATION OF RRO
C
  230 TP=PI*.5/DP
      PHIB1=ARCOS(RBC1/RLM1)
      BETAB1=PI/(2.*TG1)+(TAN(PHI)-PHI)-(TAN(PHIB1)-PHIB1)
      IF (RLM1.GE.RTF11) GO TO 285
      ARG1=((RRC1+RF1)**2 + RLM1**2 - RF1**2)/(2.*RLM1*(RRC1+RF1))
      ALPHA1=ARCOS(ARG1)
  285 RRO1=RRC1*COS(BETAB1+ALPHA1)
C
      XMIN1=RTF11*SIN(BETAB1)
      PAP1=PAP1/TODEGR
C
      PHIB2=ARCOS(RBC2/RLM2)
      BETAB2=PI/(2.*TG2)+(TAN(PHI)-PHI)-(TAN(PHIB2)-PHIB2)
      IF (RLM2.GE.RTF22) GO TO 290
      ARG2=((RRC2+RF2)**2 + RLM2**2 - RF2**2)/(2.*RLM2*(RRC2+RF2))
      ALPHA2=ARCOS(ARG2)
  290 RRO2=RRC2*COS(BETAB2+ALPHA2)
C
      XMIN2=RTF22*SIN(BETAB2)
      PAP2=PAP2/TODEGR
C
      WRITE(LU,291) BETAB1,BETAB2,ARG1,ARG2,ALPHA1,ALPHA2,RRO1,RRO2
291   FORMAT('BETAB1,BETAB2  :',2F14.7/
     2       'ARG1,ARG2      :',2F14.7/
     3       'ALPHA1,ALPHA2  :',2F14.7/
     4       'RRO1,RRO2      :',2F14.7/)
C
C
C*****CALCULATION OF INVOLUTE PROFILE COORDINATES, GEAR 1
C
      DO 330 J=1,LI1
```

```
      ET1=0.
      PE1=0.
C
      R1=RAC1-RINCI1*(FLOAT(J-1))
      PHI1=ARCOS(RBC1/R1)
      BETA1=PI/(2.*TG1) + (TAN(PHI)-PHI) - (TAN(PHI1)-PHI1)
      THETA1(J)=PHI1-BETA1
      RA1=TODEGR*TAN(PHI1)
      IF (J.EQ.1) RA1=RAT1
C
C*****CHECK FOR  TIP MODIFICATIONS
C
      IF (RATM1.EQ.0..OR.RA1.LT.RAM1) GO TO 300
      IF (STTM1.EQ.0.) ET1=PATM1*(1.-SQRT((RAT1-RA1)/RATM1))
      IF (PATM1.EQ.0.) ET1=STTM1*(RA1-RAM1)/RATM1
C
C*****CHECK FOR SINUSOIDAL ERRORS
C
300   IF (PER1.EQ.0.) GO TO 310
      IF (RA1.GT.RAM1) PE1=PER1*SIN(PAP1)
      IF (RA1.LT.RAM1)
     &    PE1=PER1*SIN((PI*(RAM1-RA1)*CYC1/RATIP1)+PAP1)
C
C*****CHECK FOR BOTTOM MODIFICATIONS
C
310   IF (RABM1.EQ.0..OR.RA1.GT.RAN1) GO TO 320
      IF (STBM1.EQ.0.) ET1=PABM1*(1.-SQRT((RA1-RABI1)/RABM1))
      IF (PABM1.EQ.0.) ET1=STBM1*(RA1-RAN1)/RABM1
C
320   X1(J)=R1*SIN(BETA1) + (ET1+PE1)/COS(THETA1(J))
      Y1(J)=R1*COS(BETA1) - RRO1
      IF (J.NE.1) THETA1(J-1)=ATAN((X1(J)-X1(J-1))/(Y1(J-1)-Y1(J)))
330   CONTINUE
C
C*****FILLET COORDINATE POINTS, GEAR 1
C
      BETA1=ATAN(X1(LI1)/(Y1(LI1)+RRO1))
      RINCB1=(R1-RRC1)/FLOAT(L1-LI1)
      LI11=LI1+1
      DO 340 J=LI11,L1
      RFIL1=R1-RINCB1*FLOAT(J-LI1)
      IF (RFIL1.GE.RTF11) ARC1=ALPHA1
      X=((RRC1+RF1)**2+RFIL1**2-RF1**2)/(2.*RFIL1*(RRC1+RF1))
      IF (X.GT.1.) X=1.0
      IF (RFIL1.LT.RTF11) ARC1=ARCOS(X)
      BETAF1=BETA1+ALPHA1-ARC1
      X1(J)=RFIL1*SIN(BETAF1)
      Y1(J)=RFIL1*COS(BETAF1) - RRO1
340   THETA1(J-1)=ATAN((X1(J)-X1(J-1))/(Y1(J-1)-Y1(J)))
      THETA1(L1)=.5*PI - BETAF1
C
C*****CALCULATION OF INVOLUTE PROFILE COORDINATES, GEAR 2
C
      DO 380 J=1,LI2
      ET2=0.
```

```
      PE2=0.
C
      R2=RAC2-RINCI2*(FLOAT(J-1))
      PHI2=ARCOS(RBC2/R2)
      BETA2=PI/(2.*TG2) + (TAN(PHI)-PHI) - (TAN(PHI2)-PHI2)
      THETA2(J)=PHI2-BETA2
      RA2=TODEGR*TAN(PHI2)
      IF (J.EQ.1) RA2=RAT2
C
      IF (RATM2.EQ.0..OR.RA2.LT.RAM2) GO TO 350
      IF (STTM2.EQ.0.) ET2=PATM2*(1.-SQRT((RAT2-RA1)/RATM2))
      IF (PATM2.EQ.0.) ET2=STTM2*(RA1-RAM2)/RATM2
C
350   IF (PER2.EQ.0.) GO TO 360
      IF (RA2.GT.RAM2) PE2=PER2*SIN(PAP2)
      IF (RA2.LT.RAM2)
     &    PE2=PER2*SIN((PI*(RAM2-RA2)*CYC2/RATIP2)+PAP2)
C
360   IF (RABM2.EQ.0..OR.RA2.GT.RAN1) GO TO 370
      IF (STBM2.EQ.0.) ET2=PABM2*(1.-SQRT((RA2-RABI2)/RABM2))
      IF (PABM2.EQ.0.) ET2=STBM2*(RA2-RAN2)/RABM2
C
370   X2(J)=R2*SIN(BETA2) + (ET2+PE2)/COS(THETA2(J))
      Y2(J)=R2*COS(BETA2) - RRO2
      IF (J.NE.1) THETA2(J-1)=ATAN((X2(J)-X2(J-1))/(Y2(J-1)-Y2(J)))
380   CONTINUE
C
C*****FILLET COORDINATE POINTS, GEAR 2
C
      RINCB2=(R2-RRC2)/FLOAT(L1-LI2)
C
      WRITE(LU,16) RLM1,RLM2,R1,R2,RINCB1,RINCB2,RTF11,RTF22
16    FORMAT('RLM1,RLM2      :',2F14.7/
     1       'R1,R2          :',2F14.7/
     2       'RINCB1,RINCB2  :',2F14.7/
     3       'RTF11,RTF22    :',2F14.7/)
C
      LI22=LI2+1
      DO 390 J=LI22,L2
      RFIL2=R2-RINCB2*FLOAT(J-LI2)
      X=((RRC2+RF2)**2+RFIL2**2-RF2**2)/(2.*RFIL2*(RRC2+RF2))
      IF (RFIL2.GE.RTF22) ARC2=ALPHA2
      IF (RFIL2.LT.RTF22)
     &ARC2=ARCOS(X)
      BETAF2=BETA2+ALPHA2-ARC2
      X2(J)=RFIL2*SIN(BETAF2)
      Y2(J)=RFIL2*COS(BETAF2) - RRO2
390   THETA2(J-1)=ATAN((X2(J)-X2(J-1))/(Y2(J-1)-Y2(J)))
      THETA2(L2)=.5*PI - BETAF2
C
C     CONTACT RATIO CALCULATIONS
C
      AUX1=ARCOS(RBC1/RTI1)
      AUX2=ARCOS(RBC2/RTI2)
      AL1=ARCOS(RBC1/RBI1)
```

```
      AL2=ARCOS(RBC2/RBI2)
      CRU1=RPC1*(SIN(PHI)-COS(PHI)*TAN(AL1))/BP
      CRU2=RPC2*(SIN(PHI)-COS(PHI)*TAN(AL2))/BP
      CR1=((RTI2)*SIN(AUX2)-RPC2*SIN(PHI))/BP
      CR2=((RTI1)*SIN(AUX1)-RPC1*SIN(PHI))/BP
C
      WRITE(LU,17) CRU1,CRU2,CR1,CR2,RBC1,RBC2,RBI1,RBI2
17    FORMAT('CRU1,CRU2      :',2F14.7/
     1         'CR1,CR2        :',2F14.7/
     2         'RBC1,RBC2      :',2F14.7/
     3         'RBI1,RBI2      :',2F14.7/)
C
      IF((RBC1.GE.RBI1).AND.(RBC2.GE.RBI2)) GO TO 18
      IF(CRU1.LE.CR1) CR1=CRU1
      IF(CRU2.LE.CR2) CR2=CRU2
      IF(CRU1.GT.CR1) CR1=CR1
      IF(CRU2.GT.CR2) CR2=CR2
   18 CR=CR1+CR2
      SP=CR1*BP
      EP=CR2*BP
      SE=CR*BP
      INF = 1
   12 FORMAT(T46,'THE THEORETICAL CONTACT RATIO =',F8.3/)
C
C*****PIT INSERTION
C
      DEEP1=0.0
      DEEP2=0.0
      IF (DEEP1.EQ.0.0) GO TO 4561
      DO 4560 I=IPIT11,IPIT12
 4560 X1(I)=X1(I)-DEEP1
 4561 IF (DEEP2.EQ.0.0) GO TO 4563
      DO 4562 I=IPIT21,IPIT22
 4562 X2(I)=X2(I)-DEEP2
 4563 CONTINUE
C
      WRITE(LU,4570) RAC1,RAC2,LI1,LI2,RF1,RF2,RRO1,RRO2
4570  FORMAT('RAC1,RAC2      :',2F14.7/
     1       'LI1,LI2        :',2I14/
     2       'RF1,RF2        :',2F14.7/
     3       'RRO1,RRO2      :',2F14.7//)
      WRITE(LU,292) PER1,PER2,CYC1,CYC2,PAP1,PAP2
  292 FORMAT('SINE ERRORS',6F11.6/)
C
C   NEW CODE FROM SLOWM FOLLOWS
C
      DO 702 I=1,L1
      W1(I)=X1(I)
      Z1(I)=RRO1+Y1(I)
      RR1(I)=SQRT(W1(I)**2+Z1(I)**2)
      IF(RR1(I).GT.RAC1) RR1(I)=RAC1
      W2(I)=X2(I)
      Z2(I)=RRO2+Y2(I)
      RR2(I)=SQRT(W2(I)**2+Z2(I)**2)
      IF(RR2(I).GT.RAC2) RR2(I)=RAC2
```

```
  702 CONTINUE
      DO 703 I=1,L1
      IPP=I
      IF(RPC1.GE.RR1(I)) GO TO 704
  703 CONTINUE
  704 CONTINUE
C
      WRITE(1,4571)
 4571 FORMAT('LIST DATA?           [Y/N] >_')
      READ(1,4572) IPICK
 4572 FORMAT(A1)
      IF (IPICK.EQ.1HN) GO TO 4610
C
      WRITE(LU,4578)
 4578 FORMAT('  I',8X,'X1',12X,'Y1',10X,'THETA1',10X,'X2',12X,'Y2',
     &10X,'THETA2')
      DO 4608 I=1,L1
 4608 WRITE(LU,4609) I,X1(I),Y1(I),THETA1(I),X2(I),Y2(I),THETA2(I)
 4609 FORMAT(I3,6F14.7)
C
 4610 WRITE(1,4620)
 4620 FORMAT('WRITE TO DATA FILE?  [Y/N] >_')
      READ(1,4572) IPICK
      IF (IPICK.EQ.1HY) CALL TRANS
C
      END
```

```
C
C
C
      SUBROUTINE TRANS
      COMMON/C1/BUF1(1200)
      COMMON/C2/BUF2(16)
      COMMON/C3/BUF3(17)
C
      DIMENSION IDCB(144),INAME(3)
      DATA INAME/2HDP,2HRO,2HFL/,ICODE/1/
C
      CALL OPEN(IDCB,IER,INAME,1,-99,36)
      IF (IER.GT.0) GO TO 13
C
9     WRITE(1,10) IER,ICODE
10    FORMAT('FMP ERROR :',I5,' AT ICODE=',I2,'        PROGRAM ABORTED')
      GO TO 50
C
13    ICODE=ICODE+1
      CALL WRITF(IDCB,IER,BUF1,2400)
      IF (IER.LT.0) GO TO 9
      ICODE=ICODE+1
      CALL WRITF(IDCB,IER,BUF2,32)
      IF (IER.LT.0) GO TO 9
      ICODE=ICODE+1
      CALL WRITF(IDCB,IER,BUF3,34)
      IF (IER.LT.0) GO TO 9
C
      WRITE(1,30) INAME
30    FORMAT('DATA FROM PROFL WAS WRITTEN TO FILE ',3A2)
C
50    CALL CLOSE(IDCB,IER)
      RETURN
      END
C
C
C
      BLOCK DATA
      COMMON/C1/BUF1(1200)
      COMMON/C2/E1,G1,PR1,YP1,RRO1,RI1,FW1,XMIN1,
     1          E2,G2,PR2,YP2,RRO2,RI2,FW2,XMIN2
      COMMON/C3/PI,PHI,TG1,TG2,TP,PD1,PD2,RPC1,RPC2,RAC1,RAC2,TIN,RBC1,
     1          RBC2,C,SP,EP
C
      DATA E1/30.E06/,E2/30.E06/,PR1/.285/,PR2/.285/
      DATA G1/.288/,G2/.288/,PI/3.141592654/
C     DATA TG1/14./,TG2/28./,RI1/.66/,RI2/1.500/
      DATA TG1/25./,TG2/24./,RI1/2.2300/,RI2/2.1300/
      DATA FW1,FW2,TIN/1.0,1.,1500.0/
      END
```

```
FTN4,L
      PROGRAM EXDEF
C
      COMMON/C1/X1(100),X2(100),Y1(100),Y2(100),THETA1(100),THETA2(100),
     1          W1(100),W2(100),Z1(100),Z2(100),R1(100),R2(100)
      COMMON/C2/E1,G1,PR1,YP1,RRO1,RI1,F1,XMIN1,
     1          E2,G2,PR2,YP2,RRO2,RI2,F2,XMIN2
      COMMON/C4/DCP1(100),DCP2(100),THETC1(100),THETC2(100)
C
      DIMENSION X(100),Y(100),A(100),DCP(100),THETA(100),BML(100)
      DIMENSION MI(100),THETAC(100),PL(100),VL(100),DM4(100)
      REAL MI1,MI2,MI
C
      WRITE(1,9000)
 9000 FORMAT('ENTER LU NUMBER OF OUTPUT DEVICE')
      READ(1,*) LU
      CALL ERLST(LU)
C
      CALL TRANR
C
      L1=100
      L2=L1
      KH=1
      KL=100
C
      PI=3.141592654
      IG=1
    5 IF(IG.EQ.3) GO TO 150
      IF(IG.EQ.2) GO TO 15
      E = E1
      G = G1
      PR = PR1
      YP = YP1
      RRO = RRO1
      RI = RI1
      XMIN = XMIN1
      F = F1
      LL = L1
      DO 10 I = 1,LL
      X(I) = X1(I)
      Y(I) = Y1(I)
      THETA(I) = THETA1(I)
      A(I) = 2.*X(I)*F
   10 MI(I)=(F*(2.*X(I))**3)/12.
      GO TO 25
   15 E = E2
      G = G2
      PR = PR2
      YP = YP2
      RRO = RRO2
      RI = RI2
      XMIN = XMIN2
      F = F2
      LL = L2
      DO 20 I = 1,LL
```

```
      X(I) = X2(I)
      Y(I) = Y2(I)
      THETA(I) = THETA2(I)
      A(I) = 2.*X(I)*F
   20 MI(I)=(F*(2.*X(I))**3)/12.
   25 BW=2.*XMIN
      IF (IG.EQ.1) YH=Y1(1)
      IF (IG.EQ.1) YL=Y1(100)
      IF (IG.EQ.2) YH=Y2(1)
      IF (IG.EQ.2) YL=Y2(100)
C
      DO 30 L = 1,LL
      KH = LL +1 - L
      IF(Y(KH).GE.YH) GO TO 40
   30 CONTINUE
   40 DO 50 KL = 1,LL
      IF(Y(KL).LE.YL) GO TO 60
   50 CONTINUE
   60 DO 70 L = 1,LL
      DCP(L) = 0.0
      THETAC(L)=0.0
   70 CONTINUE
      QQ = 1.0
      DO 110 K = KH,KL
      THETAC(K) = THETA(K)
      P = QQ*SIN(THETAC(K))
      V = QQ*COS(THETAC(K))
      YCT = Y(K) - X(K)*TAN(THETAC(K))
      BR1 = (QQ*(COS(THETAC(K)))**2)/E
      BR2 = (5.2*(YCT**2))/(BW**2) + YCT/BW
      BR3 = 1.4*(1.+((TAN(THETAC(K)))**2)/3.1)
      DCPBR = BR1*(BR2 + BR3)
      T = V*(Y(K) + RRO)
      PSIRD = (T/(4.*PI*F*G))*(1./(RI**2)-1./(RRO**2))
      RCP = SQRT((RRO + Y(K))**2 + X(K)**2)
      THETAQ = ATAN(X(K)/(RRO + Y(K)))
      ARG = ABS(THETAC(K) + THETAQ)
      DCPRD = RCP*PSIRD*COS(ARG)
      DO 80 L = 1,LL
      BML(L) = 0.0
      PL(L) = 0.0
   80 VL(L)=0.0
      DO 90 J = K,LL
      BM = (Y(K)-Y(J))*V - X(K)*P
      BML(J)=BM*BM/(MI(J)*QQ)
      PL(J)=P*P/(A(J)*QQ)
   90 VL(J)=V*V/(A(J)*QQ)
      DCPBM = 0.0
      DCPP = 0.0
      DCPV = 0.0
      N = LL - 1
      DO 100 J = K,N
      DELTAY = Y(J) - Y(J+1)
      DCPBM = DCPBM + ((BML(J)+BML(J+1))*.5)*DELTAY
      DCPP = DCPP + ((PL(J)+PL(J+1))*.5)*DELTAY
```

```
      DCPV = DCPV + ((VL(J)+VL(J+1))*.5)*DELTAY
  100 CONTINUE
      DCP(K) = DCPRD + DCPBR + DCPBM/E + DCPP/E + (1.2*DCPV)/G
      IF (K.LT.11) WRITE(LU,105) DCPRD,DCPBR,DCPBM,DCPP,DCPV
105   FORMAT(5E14.7)
  110 CONTINUE
      IF(IG.EQ.2) GO TO 120
      DO 115 I = 1,LL
      DCP1(I) = DCP(I)
  115 THETC1(I)=THETAC(I)
      IG=IG+1
      GO TO 5
  120 DO 125 I = 1,LL
      DCP2(I) = DCP(I)
  125 THETC2(I)=THETAC(I)
      IG=IG+1
      GO TO 5
C
150   CALL TRANW
C
      WRITE(1,155)
155   FORMAT('LIST DATA?  [Y/N]')
      READ(1,156) IPICK
156   FORMAT(A1)
      IF (IPICK.EQ.1HN) GO TO 159
      WRITE(LU,157)
157   FORMAT('X1',5X,'Y1',5X,'DCP1',5X,'X2',5X,'Y2',5X,'DCP2')
      WRITE(LU,158) (I,X1(I),Y1(I),DCP1(I),X2(I),Y2(I),DCP2(I),I=1,100)
158   FORMAT(I4,2F11.7,E13.7,2X,2F11.7,E13.7)
159   CONTINUE
C
  999 END
```

```
C
C
C
      SUBROUTINE TRANR
      COMMON/C1/BUF1(1200)
      COMMON/C2/BUF2(16)
C
      DIMENSION IDCB(144),INAME(3)
      DATA INAME/2HDP,2HR0,2HFL/,ICODE/1/
C
      CALL OPEN(IDCB,IER,INAME,0,-99,36)
      IF (IER.GT.0) GO TO 13
C
9     WRITE(1,10) IER,ICODE
10    FORMAT('FMP ERROR :',I5,' AT ICODE=',I2,'     PROGRAM ABORTED')
      GO TO 50
C
13    ICODE=ICODE+1
      CALL READF(IDCB,IER,BUF1)
      IF (IER.LT.0) GO TO 9
      ICODE=ICODE+1
      CALL READF(IDCB,IER,BUF2)
      IF (IER.LT.0) GO TO 9
      ICODE=ICODE+1
C
      WRITE(1,30) INAME
30    FORMAT('DATA FROM ',3A2,' WAS READ')
C
50    CALL CLOSE(IDCB,IER)
      RETURN
      END
```

```
C
C
C
      SUBROUTINE TRANW
      COMMON/C4/BUF4(400)
C
      DIMENSION IDCB(144),INAME(3)
      DATA INAME/2HDD,2HEF,2HL /,ICODE/1/
C
      CALL OPEN(IDCB,IER,INAME,0,-99,36)
      IF (IER.GT.0) GO TO 13
C
9     WRITE(1,10) IER,ICODE
10    FORMAT('FMP ERROR :',I5,' AT ICODE=',I2,'        PROGRAM ABORTED')
      GO TO 50
C
13    ICODE=ICODE+1
      CALL WRITF(IDCB,IER,BUF4,800)
      IF (IER.LT.0) GO TO 9
      ICODE=ICODE+1
C
      WRITE(1,30) INAME
30    FORMAT('DATA FROM DEFL WAS WRITTEN TO FILE ',3A2)
C
50    CALL CLOSE(IDCB,IER)
      RETURN
      END

C
C
C
      BLOCK DATA
      COMMON/C1/BUF1(1200)
      COMMON/C2/BUF2(16)
      COMMON/C4/BUF4(400)
      END
```

```
C               <830611.1023>
FTN4,L
      PROGRAM EXSTA,3
C      .
C*****THIS PROGRAM PERFORMS THE STATIC ANALYSIS OF AN EXT/EXT SPUR GEAR SET
C
      COMMON/C1/X1(100),X2(100),Y1(100),Y2(100),THETA1(100),THETA2(100),
     &          W1(100),W2(100),Z1(100),Z2(100),R1(100),R2(100)
      COMMON/C2/E1,G1,PR1,YP1,RRO1,RI1,F1,XMIN1,
     &          E2,G2,PR2,YP2,RRO2,RI2,F2,XMIN2
      COMMON/C3/PI,PHI,TG1,TG2,TP,PD1,PD2,RPC1,RPC2,RAC1,RAC2,TIN,RBC1,
     &          RBC2,C,SP,EP
      COMMON/C4/DCP1(100),DCP2(100),THETC1(100),THETC2(100)
      COMMON/C5/PSI1SL,PSI1EL,PSI2SL,PSI2EL,DELTA1,DELTA2,TDEFL1(5,50),
     &          TDEFL2(5,50)
      COMMON/C6/L1,L2,I1,IR1S,IR2E,INSTN
      COMMON/C7/BUF7(708)
      DIMENSION INAM1(3),INAM2(3),INAM3(3)
C
C
      ARCOS(X)=PI/2.-ATAN(X/SQRT(1-X**2))
      DATA INAM1/2HPA,2HRT,2H1 /,INAM2/2HPA,2HRT,2H2 /
      DATA IN4M3/2HPA,2HRT,2H3 /
C
      WRITE(1,998)
998   FORMAT('LU FOR OUTPUT DEVICE?')
      READ(1,*) LU
C
7050  CALL TRANR
      TIN=1500.0
C     GO TO 9989
C
C
      IBYPSS=1
      L1=100
      L2=L1
      PSI1SL=0.
      BUF7(1)=0.
C
      PAS=7
      LINF = 1
      P =  TIN/RBC1
      F = F1
      IF(F2.LT.F1) F = F2
C
      WRITE(1,99)
   99 FORMAT('ENTER NUMBER OF ITERATIONS  [1 - RIGID; 2 - DEFLECTED]')
      READ(1,*) ITER
      DO 120 I1=1,ITER
      WRITE(LU,9900)
9900  FORMAT(/'CALL EXEC FOR PART1')
C
      ASSIGN 100 TO INSTN
      CALL EXEC(8,INAM1)
100   CONTINUE
```

```
C
      WRITE(LU,9910)
9910  FORMAT(/'CALL EXEC FOR PART2')
C
      ASSIGN 110 TO INSTN
      CALL EXEC(8,INAM2)
110   CONTINUE
C
120   CONTINUE
C
      WRITE(LU,9930)
9930  FORMAT(/'CALL EXEC FOR PART3?  [Y/N] > _ ')
      READ(1,130) IPICK
  130 FORMAT(A1)
      IF (IPICK.EQ.1HY) CALL EXEC(8,INAM3)
C
      END
```

```
C
C
C
      SUBROUTINE TRANR
      COMMON/C1/BUF1(1200)
      COMMON/C2/BUF2(16)
      COMMON/C3/BUF3(17)
      COMMON/C4/BUF4(400)
C
      DIMENSION IDCB(144),INAM1(3),INAM2(3)
      DATA INAM1/2HDP,2HRO,2HFL/,ICODE/1/
      DATA INAM2/2HDD,2HEF,2HL /
C
      CALL OPEN(IDCB,IER,INAM1,0,-99,36)
      IF (IER.GT.0) GO TO 30
C
10    WRITE(1,20) IER,ICODE
20    FORMAT('FMP ERROR :',I5,' AT ICODE=',I2,'     PROGRAM ABORTED')
      GO TO 50
C
30    ICODE=ICODE+1
      CALL READF(IDCB,IER,BUF1)
      IF (IER.LT.0) GO TO 10
      ICODE=ICODE+1
      CALL READF(IDCB,IER,BUF2)
      IF (IER.LT.0) GO TO 10
      ICODE=ICODE+1
      CALL READF(IDCB,IER,BUF3)
      IF (IER.LT.0) GO TO 10
      ICODE=ICODE+1
C
      WRITE(1,40) INAM1
40    FORMAT('DATA FROM ',3A2,' WAS READ')
C
50    CALL CLOSE(IDCB,IER)
C
      CALL OPEN(IDCB,IER,INAM2,0,-99,36)
      IF (IER.GT.0) GO TO 70
C
60    WRITE(1,20) IER,ICODE
      GO TO 90
C
70    ICODE=ICODE+1
      CALL READF(IDCB,IER,BUF4)
      IF (IER.LT.0) GO TO 60
      ICODE=ICODE+1
C
      WRITE(1,40) INAM2
80    FORMAT('DATA FROM ',3A2,' WAS READ')
C
90    CALL CLOSE(IDCB,IER)
      RETURN
      END
```

```
C
C
C
      PROGRAM PART1,5
      COMMON/C1/X1(100),X2(100),Y1(100),Y2(100),THETA1(100),THETA2(100),
     &          W1(100),W2(100),Z1(100),Z2(100),R1(100),R2(100)
      COMMON/C2/E1,G1,PR1,YP1,RRO1,RI1,F1,XMIN1,
     &          E2,G2,PR2,YP2,RRO2,RI2,F2,XMIN2
      COMMON/C3/PI,PHI,TG1,TG2,TP,PD1,PD2,RPC1,RPC2,RAC1,RAC2,TIN,RBC1,
     &          RBC2,C,SP,EP
      COMMON/C4/DCP1(100),DCP2(100),THETC1(100),THETC2(100)
      COMMON/C5/PSI1SL,PSI1EL,PSI2SL,PSI2EL,DELTA1,DELTA2,TDEFL1(5,50),
     &          TDEFL2(5,50)
      COMMON/C6/L1,L2,I1,IR1S,IR2E,INSTN
      DIMENSION  U1(100),U2(100),V1(100),V2(100),CDEF(50)
      ARCOS(X)=PI/2.-ATAN(X/SQRT(1-X**2))
C
      AB=SQRT(RAC1**2-RBC1**2)+SQRT(RAC2**2-RBC2**2)-C*SIN(PHI)
      E1B=SQRT(RAC1**2-RBC1**2)
      E1A=E1B-AB
      E1P=RPC1*SIN(PHI)
      AP=E1P-E1A
      PB=AB-AP
      SP=AP
      EP=PB
C
      WRITE(1,279)
279   FORMAT('ENTER OUTPUT DEVICE')
      READ(1,*) LU
C
      IF (I1.EQ.2) GO TO 47
C
      WRITE(LU,280) SP,EP
280   FORMAT('SP :',F12.7,3X,'EP :',F12.7)
      L1=100
      L2=L1
      US = -(SP)*COS(PHI)
      VS = -(SP)*SIN(PHI) + RPC1
      UE = EP*COS(PHI)
      VE = EP*SIN(PHI) + RPC1
      R1S = SQRT(US**2 + VS**2)
      R2S=SQRT(US**2+(C-VS)**2)
      R1E=SQRT(UE**2+VE**2)
      R2E = SQRT(UE**2 + (C-VE)**2)
      ARG1 = ARCOS(RBC1/R1S)
      BETA1S = TP/PD1 + (TAN(PHI)-PHI) - (TAN(ARG1)-ARG1)
      ARG2 = ABS(US)/VS
      PSI1SL  =  PI/2. + ATAN(ARG2) + BETA1S
      ARG1 = ARCOS(RBC1/R1E)
      BETA1E = TP/PD1 + (TAN(PHI)-PHI) - (TAN(ARG1)-ARG1)
      ARG2 = VE/UE
      PSI1EL = ATAN(ARG2) + BETA1E
      ARG1 = ARCOS(RBC2/R2S)
      BETA2S = TP/PD2 + (TAN(PHI)-PHI) - (TAN(ARG1)-ARG1)
      ARG2 = ABS(US)/(C-VS)
```

```
      PSI2SL = (3.*PI)/2. - ATAN(ARG2) + BETA2S
      ARG1 = ARCOS(RBC2/R2E)
      BETA2E = TP/PD2 + (TAN(PHI)-PHI) - (TAN(ARG1)-ARG1)
      ARG2 = UE/(C-VE)
      PSI2EL = (3.*PI)/2. + ATAN(ARG2) + BETA2E
      DELTA1 = (PSI1SL - PSI1EL)/49.
      DELTA2 = (TG1/TG2)*DELTA1
C
      Y1S=R1S*(COS(BETA1S))-RR01
      Y1E=R1E*(COS(BETA1E))-RR01
      Y2S=R2S*(COS(BETA2S))-RR02
      Y2E=R2E*(COS(BETA2E))-RR02
C
      DO 290 IR1S=1,L1
      IF(R1S.GE.R1(IR1S)) GO TO 291
  290 CONTINUE
  291 DO 292 IR2S=1,L2
      IF(R2S.GE.R2(IR2S)) GO TO 293
  292 CONTINUE
  293 DO 294 IR1E=1,L1
      IF(R1E.GE.R1(IR1E)) GO TO 295
  294 CONTINUE
  295 DO 296 IR2E=1,L2
      IF(R2E.GE.R2(IR2E)) GO TO 297
  296 CONTINUE
  297 CONTINUE
      WRITE(LU,9000) IR1S,Y1S,IR2S,Y2S,IR1E,Y1E,IR2E,Y2E
 9000 FORMAT(//'THEORETICAL INITIAL AND FINAL POINTS OF CONTACT'
     & /1X,4(I8,F11.7,4X))
      WRITE(LU,8900) R1S,R1(IR1S),R2S,R2(IR2S),R1E,R1(IR1E),R2E,R2(IR2E)
 8900 FORMAT('THEORETICAL AND ACTUAL RADII TO CONTACT POINTS'/1X,
     & 4(2F11.7,4X))
47    CONTINUE
      PSS1SL=PSI1SL
      PSS1EL=PSI1EL
      PSS2EL=PSI2EL
      PSS2SL=PSI2SL
      INT=50
      WRITE(LU,9001) PSI1SL,PSI1EL,PSI2SL,PSI2EL,DELTA1,DELTA2
 9001 FORMAT('THEORETICAL INITIAL AND FINAL ANGLES OF CONTACT',
     &', ANGULAR INCREMENTS'/4F11.7,3X,2F11.7)
      V1SP=0.0
      U1SP=0.0
      V2SP=0.0
      U2SP=0.0
      V1EP=0.0
      U1EP=0.0
      V2EP=0.0
      U2EP=0.0
      U11=0.0
      U22=0.0
      U12EP=0.0
      U12SP=0.0
      MMC=0
      NNC=0
```

```
      KKKK=(2*L1)+10
      DELT=0.2
      NLIM=75
      MLIM=NLIM
      NLIM1=2*NLIM
      MLIM1=NLIM1
      L11=L1-1
      L22=L2-1
      IRDEL=30
      IR1S1=IR1S-IRDEL
      IR1S2=IR1S+IRDEL
      P1SL=PSS1SL+DELTA1*(FLOAT(NLIM-1))*DELT
      P2SL=PSS2SL-DELTA2*(FLOAT(NLIM-1))*DELT
      DIF1=(Y1(1)-Y1(L1))/(FLOAT(L1))
      DIF2=(Y2(1)-Y2(L2))/(FLOAT(L2))
      IF(DIF1.GE.DIF2) DIFF=DIF1
      IF(DIF2.GT.DIF1) DIFF=DIF2
      IF(IR1S1.LE.1) IR1S1=1
      IF(IR1S2.GT.L1) IR1S2=L1-1
      DIFF=9.99
      WRITE(LU,9017) TDEFL1(3,1),TDEFL2(3,1),TDEFL1(3,50),TDEFL2(3,50)
 9017 FORMAT('DEFL. ADDED AT ENTRANCE',2X,2F11.7/
     &       'DEFL. ADDED AT EGRESS  ',2X,2F11.7)
   49 CONTINUE
      IF (NNC.EQ.8.OR.MMC.EQ.5) DIFF=DIFF+(DIFF/3.)
      WRITE(LU,4900) PSS1SL,P1SL,PSS2SL,P2SL
 4900 FORMAT(/'PSS1SL,P1SL',2F14.7/'PSS2SL,P2SL',2F14.7)
      DIST=100.
      DO 50 N1=1,NLIM1
      PSI1SL=P1SL-DELTA1*(FLOAT(N1-1))*DELT
      PSI2SL=P2SL+DELTA2*(FLOAT(N1-1))*DELT
      DO 51 L=1,25
      LC=L
      U2(L)=Z2(L)*SIN(PSI2SL-1.5*PI)-(W2(L)+TDEFL2(3,1))*COS(PSI2SL-1.5*
     &PI)
      V2(L)=C-(Z2(L)*COS(PSI2SL-1.5*PI)+(W2(L)+TDEFL2(3,1))*SIN(PSI2SL-
     &1.5*PI))
      U2SP=U2(LC)
      V2SP=V2(LC)
      DO 52 J=IR1S1,IR1S2
      JC=J
      U1(J)=(W1(J)+TDEFL1(3,1))*SIN(PSI1SL)+Z1(J)*COS(PSI1SL)
      U1(J+1)=(W1(J+1)+TDEFL1(3,1))*SIN(PSI1SL)+Z1(J+1)*COS(PSI1SL)
      V1(J)=-(W1(J)+TDEFL1(3,1))*COS(PSI1SL)+Z1(J)*SIN(PSI1SL)
      IF (V1(J).LT.V2SP) GO TO 51
      V1(J+1)=-(W1(J+1)+TDEFL1(3,1))*COS(PSI1SL)+Z1(J+1)*SIN(PSI1SL)
      IF (V1(J).GE.V2SP.AND.V1(J+1).GT.V2SP) GO TO 52
      V12SP=V1(J)-V2SP
      ARG11=V1(J+1)-V1(J)
      ARGV11=(V2SP-V1(J))*(U1(J+1)-U1(J))
      U11=(ARGV11/ARG11)+U1(J)
      U12SP=U11-U2SP
      MMC=8
C     WRITE(LU,5200) J,L,U12SP,PSI1SL
 5200 FORMAT(2I7,2F14.7)
```

```
      IF (U11.GE.U2SP.AND.ABS(V12SP).LE.DIFF.AND.U12SP.LE.0.000100)
     &  GO TO 53
      MMC=7
      IF (ABS(U12SP).LE.0.000010.AND.ABS(V12SP).LE.DIFF) GO TO 53
      IF (ABS(U12SP).GT.ABS(DIST)) GO TO 52
      DIST=U12SP
      JCA=J
      LCA=L
      ANGS1=PSI1SL
      ANGS2=PSI2SL
   52 CONTINUE
   51 CONTINUE
   50 CONTINUE
      WRITE(LU,5000) DIST,JCA,LCA,ANGS1,ANGS2
 5000 FORMAT(/'MISSED INITIAL CONTACT POINT'/
     &'CLOSEST APPROACH IS',F11.7,' BETWEEN PTS.',I4,' AND',I4/
     &' AT PS1I=',F10.7,'   PSI2=',F10.7)
      MMC=5
   53 CONTINUE
      WRITE(LU,9010) N1,J,L,U11,V1(J),U2(L),V2(L),U12SP,V12SP,MMC
 9010 FORMAT('ACTUAL START OF CONTACT'/1X,
     & 3I5,3(2F11.7,4X),15X,'CONTACT CODE -',I3)
 4910 FORMAT('N1,J,L,MMC,-',4I5,20X,'U11,V12SP,U12SP-',3F14.7,/,
     &' U1(J),U2(L),V1(J),V2(L)-',4F14.7,20X,'PSS1SL,PSS2SL-',2F14.7)
      Y1S=Z1(JC)-RR01
      Y2S=Z2(LC)-RR02
      V1SP=V2SP
      U1SP=U2SP
      P1EL=PSS1EL-DELTA1*(FLOAT(MLIM-1))*DELT
      P2EL=PSS2EL+DELTA2*(FLOAT(MLIM-1))*DELT
      IR2E1=IR2E-IRDEL
      IR2E2=IR2E+IRDEL
      IF(IR2E1.LE.1) IR2E1=1
      IF(IR2E2.GT.L2) IR2E2=L2-1
      WRITE(LU,4920) PSS1EL,P1EL,PSS2EL,P2EL
 4920 FORMAT(/'PSS1EL,P1EL',2F14.7/'PSS2EL,P2EL',2F14.7)
      DO 60 M1=1,MLIM1
      PSI1EL=P1EL+DELTA1*(FLOAT(M1-1))*DELT
      PSI2EL=P2EL-DELTA2*(FLOAT(M1-1))*DELT
      DO 61 J=1,25
      JE=J
      U1(J)=(W1(J)+TDEFL1(3,INT))*SIN(PSI1EL)+Z1(J)*COS(PSI1EL)
      V1(J)=-(W1(J)+TDEFL1(3,INT))*COS(PSI1EL)+Z1(J)*SIN(PSI1EL)
      U1EP=U1(JE)
      V1EP=V1(JE)
      DO 62 L=IR2E1,IR2E2
      LE=L
      U2(L)=Z2(L)*SIN(PSI2EL-1.5*PI)-(W2(L)+TDEFL2(3,INT))*COS(PSI2EL
     &-1.5*PI)
      U2(L+1)=Z2(L+1)*SIN(PSI2EL-1.5*PI)-(W2(L+1)+TDEFL2(3,INT))*
     &COS(PSI2EL-1.5*PI)
      V2(L)=C-(Z2(L)*COS(PSI2EL-1.5*PI)+(W2(L)+TDEFL2(3,INT))*SIN(PSI2EL
     &-1.5*PI))
      IF (V2(L).GT.V1EP) GO TO 61
      V2(L+1)=C-(Z2(L+1)*COS(PSI2EL-1.5*PI)+(W2(L+1)+TDEFL2(3,INT))*SIN
```

```
     &(PSI2EL-1.5*PI))
      IF (V2(L).LE.V1EP.AND.V2(L+1).LT.V1EP) GO TO 62
      ARG22=V2(L+1)-V2(L)
      ARGV22=(V1EP-V2(L))*(U2(L+1)-U2(L))
      U22=(ARGV22/ARG22)+U2(L)
      U12EP=U1EP-U22
      V12EP=V2(L)-V1EP
      NNC=5
      IF (U1EP.GE.U22.AND.ABS(V12SP).LE.DIFF.AND.U12EP.LE.0.000080)
     &  GO TO 63
      NNC=6
      IF (ABS(U12EP).LE.0.000010.AND.ABS(V12EP).LE.DIFF) GO TO 63
   62 CONTINUE
   61 CONTINUE
   60 CONTINUE
      NNC=8
   63 CONTINUE
      WRITE(LU,9020) M1,J,L,U1(J),V1(J),U22,V2(L),U12EP,V12EP,DPSI2L,
     & NNC
 9020 FORMAT('ACTUAL END OF CONTACT',/1X,3I5,3(2F11.7,4X),F11.7,
     &4X,'CONTACT CODE -',I3)
      DELTA1=(PSI1SL-PSI1EL)/49.
      DELTA2=(TG1/TG2)*DELTA1
      Y1E=Z1(JE)-RR01
      Y2E=Z2(LE)-RR02
      V2EP=V1EP
      U2EP=U1EP
      R1S=SQRT(U1SP**2+V1SP**2)
      R1E=SQRT(U1EP**2+V1EP**2)
   88 CONTINUE
      IF (NNC.EQ.8.OR.MMC.EQ.5) GO TO 49
      DELTA1 = (PSI1SL - PSI1EL)/(INT-1)
      DELTA2 = (TG1/TG2)*DELTA1
      PRANG=ATAN(ABS(R1E*SIN(PSI1EL)-R1S*SIN(PSI1SL))/ABS(R1E*
     &COS(PSI1EL)-R1S*COS(PSI1SL)))*180./PI
      WRITE(LU,9021) PSI1SL,PSI1EL,PSI2SL,PSI2EL,DELTA1,DELTA2
 9021 FORMAT(/'ACTUAL INITIAL AND FINAL ANGLES OF CONTACT',
     &', ANGULAR INCREMENTS'/4F11.7,3X,2F11.7)
      WRITE(LU,9030) MMC,NNC,PRANG
 9030 FORMAT('CONTACT CODES:',2I6/'PRESSURE ANGLE:',F6.2)
C
9989  GO TO INSTN
9990  CONTINUE
      END
```

```
C
C
C
C
      PROGRAM PART2,5
      COMMON/C1/X1(100),X2(100),Y1(100),Y2(100),THETA1(100),THETA2(100),
     &          W1(100),W2(100),Z1(100),Z2(100),R1(100),R2(100)
      COMMON/C2/E1,G1,PR1,YP1,RRO1,RI1,F1,XMIN1,
     &          E2,G2,PR2,YP2,RRO2,RI2,F2,XMIN2
      COMMON/C3/PI,PHI,TG1,TG2,TP,PD1,PD2,RPC1,RPC2,RAC1,RAC2,TIN,RBC1,
     &          RBC2,C,SP,EP
      COMMON/C4/DCP1(100),DCP2(100),THETC1(100),THETC2(100)
      COMMON/C5/PSI1SL,PSI1EL,PSI2SL,PSI2EL,DELTA1,DELTA2,TDEFL1(5,50),
     &          TDEFL2(5,50)
      COMMON/C6/L1,L2,I1,IR1S,IR2E,INSTN
      COMMON/C7/VELR(50),PSI1(50),PSI2(50),Q(5,50),TPS(5,50),CMS(50),
     &          ANG(8)
C
      DIMENSION  U1(100),U2(100),V1(100),V2(100),CDEF(50),
     &RESIDL(2,50),THUVP(50),RC1(5,50),
     &RC2(5,50),RCC1(5,50),RCC2(5,50),RNCP(50),
     &THBUV(50),TDEF1(50),TDEF2(50),HDEF(50),SVS(50),
     &XC1(5,50),XC2(5,50),YC1(5,50),YC2(5,50),PHIOP(50),
     &PSI(50),HZPS(50),PVS(50),VELRAT(100),STATLD(100)
      REAL M,JD,JG1,JG2,JL,KDS,KGPAVG,KLS,LDS,LLS,LLR,
     &MI1,MI2,KT
      DIMENSION PSI1P(5),PSI2P(5),UCP(5),VCP(5),XCP1(5),
     &XCP2(5),YCP1(5),YCP2(5),RCP1(5),RCP2(5),RCCP1(5),
     &RCCP2(5),UC1(5),UC2(5),VC1(5),VC2(5),TDCP1(5),
     &TDCP2(5),THCP1(5),THCP2(5),PSIRD1(5),PSIRD2(5),
     &H1(5),H2(5),HD(5),CDEFL(5),STIFF(5),QTP(5),PSI1TP(5),
     &PSI2TP(5),ZCP1(5),ZCP2(5),WCP1(5),WCP2(5),TD(5)
      DIMENSION ITIME(5),IPT1(5),IPT2(5)
      DATA IR1S2,IR2E2/70,72/, RPMIN,RPMOUT/30.,10./
      ARCOS(X)=PI/2.-ATAN(X/SQRT(1-X**2))
C
      WRITE(1,9000)
9000  FORMAT('ENTER LU OF OUTPUT DEVICE AND NO. OF ANGULAR INTERVALS')
      READ(1,*) LU,INT
      CALL EXEC(11,ITIME)
      WRITE(LU,9001) ITIME(5),ITIME(4),ITIME(3),ITIME(2),ITIME(1)
9001  FORMAT(/'START PART2 EXEC',I5,4(':',I2))
      WRITE(LU,9002) PSI1SL,PSI1EL,PSI2SL,PSI2EL,DELTA1,DELTA2,TG1,TG2
9002  FORMAT('PSI1SL,PSI1EL',2F11.7,3X,'PSI2SL,PSI2EL',2F11.7/
     &       'DELTA1,DELTA2',2F11.7,3X,'TG1,TG2',2F6.1)
      WRITE(1,9003)
9003  FORMAT('PRINT RESULTS? [Y/N]')
      READ(1,9004) IPICK
9004  FORMAT(A1)
C
C
      L1=100
      L2=L1
      P=TIN/RBC1
      F=F1
```

```
      C2=(1.-PR1**2)/E1 + (1.-PR2**2)/E2
      C3=(2.*(1.-PR1**2))/(PI*F1*E1)
      C4=PR1/(2.*(1.-PR1))
      C5=(2.*(1.-PR2**2))/(PI*F2*E2)
      C6=PR2/(2.*(1.-PR2))
      PSI1EP=0.
      PSI2EP=0.
      DO 8 I = 1,50
      THUVP(I)=0.0
      VELR(I)=0.0
      THBUV(I)=0.0
      RNCP(I) = 0
      TDEF1(I) = 0.0
      TDEF2(I) = 0.0
      HDEF(I) = 0.0
      CDEF(I) = 0.0
      RESIDL(1,I)=0.0
      RESIDL(2,I)=0.0
      DO 8 K = 1,5
      Q(K,I) = 0.0
      YC1(K,I) = 0.0
      XC1(K,I) = 0.0
      YC2(K,I) = 0.0
      XC2(K,I) = 0.0
      RC1(K,I) = 0.0
      RC2(K,I) = 0.0
      RCC1(K,I) = 0.0
      RCC2(K,I) = 0.0
      TPS(K,I) = 0.0
      IPT1(K)=0
      IPT2(K)=0
    8 CONTINUE
      PSI1TP(1)=PSI1SL - 4.*PI/TG1
      PSI2TP(1)=PSI2SL + 4.*PI/TG2
      PSI1L = PSI1TP(1)
      PSI2L = PSI2TP(1)
      PSI1LS = PSI1SL
      PSI1LE = PSI1EL
      PSI1LS = 1.0001*PSI1SL
      PSI1LE = .99999*PSI1EL
      WRITE(LU,9005) TIN,P,C,L1,L2
 9005 FORMAT('TIN',F9.2,'  P',F9.2,'  C',F9.5,'  LI,L2',2I5)
      DO 46 I = 1,INT
      IF (IPICK.EQ.1HY) WRITE(LU,461) I
  461 FORMAT(' ANGULAR POSITION',I3)
      NCTP=0
      V11=0.0
      DO 10 K = 2,5
      J = K - 1
      PSI1TP(K)=PSI1TP(J) + 2.*PI/TG1
      PSI2TP(K)=PSI2TP(J) - 2.*PI/TG2
   10 CONTINUE
      PSI1(I)=PSI1TP(3)
      PSI2(I)=PSI2TP(3)
      DO 23 K = 1,5
```

```
      PSI11=PSI1TP(K)
      PSI22=PSI2TP(K)
      UCP(K) = 0.0
      VCP(K) = 0.0
      ZCP1(K) = 0.0
      ZCP2(K) = 0.0
      WCP1(K) = 0.0
      WCP2(K) = 0.0
      XCP1(K) = 0.0
      XCP2(K) = 0.0
      YCP1(K) = 0.0
      YCP2(K) = 0.0
      RCP1(K) = 0.0
      RCP2(K) = 0.0
      RCCP1(K)=0.0
      RCCP2(K)=0.0
      JCPT1=0
      JCPT2=0
      JCPB1=0
      JCPB2=0
      VEL1=0.0
      VEL2=0.0
      IF((PSI1TP(K).GT.PSI1LS).OR.(PSI1TP(K).LT.PSI1LE)) GO TO 23
      IR1S22=IR1S2+1
      IPRIME=0
      IF (K.EQ.3) GO TO 8990
      ARG=(PSI1TP(K)-PSI1TP(3))/DELTA1
      IPRIME=IFIX(ARG)+1
      IF (ARG.LT.0) IPRIME=IFIX(ARG)-1
8990  N=I-IPRIME
      IF ((I-IPRIME).LE.0.OR.(I-IPRIME).GT.50) GO TO 23
      REMEM1=TDEFL1(3,N)
      REMEM2=TDEFL2(3,N)
      IF (K.LE.3) GO TO 8993
      TDEFL1(3,N)=0.0
      TDEFL2(3,N)=0.0
      IF (I1.EQ.1) GO TO 8993
      TDEFL1(3,N)=RESIDL(1,N)
      TDEFL2(3,N)=RESIDL(2,N)
8993  CONTINUE
      DO 901 J=1,IR1S22
      U1(J)=(W1(J)+TDEFL1(3,N))*SIN(PSI1TP(K))+Z1(J)*COS(PSI1TP(K))
      V1(J)=-(W1(J)+TDEFL1(3,N))*COS(PSI1TP(K))+Z1(J)*SIN(PSI1TP(K))
 901  CONTINUE
      IR2E22=IR2E2+1
      DO 902 L=1,IR2E22
      U2(L)=Z2(L)*SIN(PSI2TP(K)-1.5*PI)-(W2(L)+TDEFL2(3,N))*COS(PSI2TP(K
     &)-1.5*PI)
      V2(L)=C-((Z2(L)*COS(PSI2TP(K)-1.5*PI)+(W2(L)+TDEFL2(3,N))*SIN(PSI2
     &TP(K)-1.5*PI)))
 902  CONTINUE
      TDEFL1(3,N)=REMEM1
      TDEFL2(3,N)=REMEM2
  11  CONTINUE
      MMM=91
```

```
C THIS SEGMENT LOCATES THE CONTACT POINT BETWEEN A GIVEN TOOTH PAIR
      IF((PSI1TP(K).GT.PSI1LS).OR.(PSI1TP(K).LT.PSI1LE)) GO TO 23
      DISTL = (RAC1 - RRC1)/(FLOAT(L1)*5.)
      GO TO 14
   13 DISTL = 2.*DISTL
   14 DO 15 J=1,IR1S2
      DIST = (ABS(TAN(PHI)*U1(J)-V1(J)+RPC1))/(SQRT((TAN(PHI))**2+1.))
      IF(DIST.LT.DISTL) GO TO 16
   15 CONTINUE
      GO TO 13
   16 IF(DIST.EQ.0.0) GO TO 21
      IF(J.EQ.1) GO TO 18
      IF(U1(J).EQ.0.0) GO TO 17
      ARGS = ABS(RPC1-V1(J))/ABS(U1(J))
      SLOPE = ATAN(ARGS)
      IF((SLOPE.GT.PHI).AND.(U1(J).LT.0.0)) GO TO 19
      IF((SLOPE.LT.PHI).AND.(U1(J).GT.0.0)) GO TO 19
      GO TO 18
   17 IF(V1(J).LE.RPC1) GO TO 19
C POINT IS ABOVE THE LINE OF ACTION
   18 UA = U1(J)
      VA = V1(J)
      UB = U1(J+1)
      VB = V1(J+1)
      GO TO 20
C POINT IS BELOW THE LINE OF ACTION
   19 UA = U1(J-1)
      VA = V1(J-1)
      UB = U1(J)
      VB = V1(J)
   20 SLOPE = (VA-VB)/(UA-UB)
      A11 = TAN(PHI)
      A12 = RPC1
      A21 = SLOPE
      A22 = VB - SLOPE*UB
      UCP(K) = (A22-A12)/(A11-A21)
      VCP(K) = A11*UCP(K) + A12
      GO TO 22
   21 UCP(K) = U1(J)
      VCP(K) = V1(J)
   22 CONTINUE
      RCP2N = SQRT((UCP(K)**2) + ((C-VCP(K))**2))
      JCP=J
      JDEL=40
      JCPB1=JCP+JDEL
      JCPT1=JCP-JDEL
      IF (JCPB1.GE.IR1S2) JCPB1=IR1S2
      IF(JCPT1.LE.1) JCPT1=1
      R2N=SQRT((C-VCP(K))**2+UCP(K)**2)
      DO 165 L=1,IR2E22
      LCP=L
      IF(R2N.GE.R2(L)) GO TO 166
  165 CONTINUE
  166 JCPB2=LCP+JDEL
      JCPT2=LCP-JDEL
```

```
      IF (JCPB2.GE.IR2E2) JCPB2=IR2E2
      IF(JCPT2.LE.1) JCPT2=1
C
C     WRITE(LU,167) I,K,JCP,LCP,W1(JCP),Z1(JCP),W2(LCP),Z2(LCP)
167   FORMAT(/'POSITION',I3,5X,'PAIR',I2,5X,'CANDIDATES',2I5/
     & 'LOCAL GEAR COOR.   NO. 1',2F10.7,' NO.2',2F10.7)
      MMM=92
C
      U1U2P=U1(JCP)-U2(LCP)
      V1V2P=V1(JCP)-V2(LCP)
      IF((K.EQ.3).AND.(I.EQ.1).AND.(V2SP.NE.0.0)) VCP(K)=V2SP
      IF((K.EQ.3).AND.(I.EQ.1).AND.(V2SP.NE.0.0)) UCP(K)=U2SP
      IF((K.EQ.3).AND.(I.EQ.1).AND.(V2SP.NE.0.0)) GO TO 350
      IF((K.EQ.3).AND.(I.EQ.50).AND.(V1EP.NE.0.0)) VCP(K)=V1EP
      IF((K.EQ.3).AND.(I.EQ.50).AND.(V1EP.NE.0.0)) UCP(K)=U1EP
      IF((K.EQ.3).AND.(I.EQ.50).AND.(V1EP.NE.0.0)) GO TO 350
      DV11=0.0
      U11=0.0
      U22=0.0
      V11=0.0
      ALIMIT=0.00001*(1.+.35*FLOAT(2-I1))
      DO 204 J=JCPT1,JCPB1
      IF (J.EQ.JCPT1) COMPAR=100.0
      U11=U1(J)
      V11=V1(J)
      VCPT2=V2(JCPT2)-5.
      IF (V11.LT.VCPT2) MMM=3
      IF (V11.LT.VCPT2) GO TO 199
      DO 202 L=JCPT2,JCPB2
      IF (V2(L+1).GE.V11) GO TO 222
  202 CONTINUE
      MMM=7
      GO TO 204
  222 ARGV2=V2(L+1)-V2(L)
      IF (ARGV2.NE.0.0) U22=((V11-V2(L))*(U2(L+1)-U2(L))/ARGV2)+U2(L)
      IF (ARGV2.EQ.0.0) U22=U2(L)
      U12=U11-U22
      IF (U11.GE.U22) MMM=0
      IF (U11.GE.U22) GO TO 349
      IF(ABS(U12).LE.ALIMIT) MMM=18
      IF(ABS(U12).LE.ALIMIT) GO TO 349
      MCA=1
      IF (V1(J+1).GT.V2(L)) GO TO 203
      ARGV3=V1(J+1)-V1(J)
      IF(ARGV3.NE.0.0) U11=((V2(L)-V1(J))*(U1(J+1)-U1(J))/ARGV3)+U1(J)
      IF (ARGV3.EQ.0.0) U11=U1(J+1)
      U12=U11-U2(L)
      IF (U11.GE.U2(L)) MMM=72
      IF (U11.GE.U2(L)) GO TO 349
      IF(ABS(U12).LE.ALIMIT) MMM=73
      IF(ABS(U12).LE.ALIMIT) GO TO 349
      MCA=2
      IF (L.EQ.1) GO TO 203
      IF (V1(J+1).GT.V2(L-1)) GO TO 203
      IF (ARGV3.NE.0.0) U11=((V2(L-1)-V1(J))*(U1(J+1)-U1(J))/(V1(J+1)-
```

```
     &V1(J)))+U1(J)
      IF (ARGV3.EQ.0.0) U11=U1(J+1)
      U12=U11-U2(L-1)
      IF (U11.GE.U2(L-1)) MMM=78
      IF (U11.GE.U2(L-1)) GO TO 349
      IF(ABS(U12).LE.ALIMIT) MMM=79
      IF(ABS(U12).LE.ALIMIT) GO TO 349
      MCA=3
  203 IF (ABS(U12).GE.ABS(COMPAR)) GO TO 205
      COMPAR=U12
      JCA=J
      LCA=L
205   CONTINUE
  204 CONTINUE
  199 CONTINUE
  504 CONTINUE
      WRITE(LU,505) I,K,MCA,PSI1TP(K),PSI2TP(K),COMPAR,JCA,LCA
  505 FORMAT('AT POSITION',I3,' TOOTH PAIR',I2/
     &'     NO CONTACT FOUND    MCA',I4,' PSI1,2',2F12.7/
     &'     CLOSEST APPROACH :',F12.7,' BETWEEN PTS.',I3' AND',I3)
      UC1(K)=U1(JCP)
      VC1(K)=V1(JCP)
      UC2(K)=U2(LCP)
      VC2(K)=V2(LCP)
      UVDCP=0.0
      MMM=15
      VELR(I)=TG2/TG1
      GO TO 360
  349 CONTINUE
      UCP(K)=U11
      VCP(K)=V11
  350 CONTINUE
C
C     WRITE(LU,9080) J,U1(J),V1(J),PSI1TP(K),L,U2(L),V2(L),PSI2TP(K),
C    &           MMM,U12,COMPAR
9080  FORMAT(5X,'ACTUAL',I5,3F13.7/11X,I5,3F13.7/
     &       5X,'CONTACT CODE',I5,5X,'U12,COMPAR',2F12.7)
C
      IPT1(K)=J
      IPT2(K)=L
      UVDCP=0.0
      NCTP = NCTP +1
      RCP1(K) = SQRT((UCP(K)**2) + (VCP(K)**2))
      RCP2(K)=SQRT(UCP(K)**2+(C-VCP(K))**2)
      ZCP2(K) = (C-VCP(K))*COS(PSI2TP(K)-(3.*PI)/2.) +
     1UCP(K)*SIN(PSI2TP(K)-(3.*PI)/2.)
      ZCP1(K) = UCP(K)*COS(PSI1TP(K)) + VCP(K)*SIN(PSI1TP(K))
      WCP1(K) = UCP(K)*SIN(PSI1TP(K)) - VCP(K)*COS(PSI1TP(K))
      WCP2(K) = (C-VCP(K))*SIN(PSI2TP(K)-(3.*PI)/2.) -
     1UCP(K)*COS(PSI2TP(K)-(3.*PI)/2.)
      RCCP1(K)=SQRT(ABS(RCP1(K)**2-RBC1**2))
      RCCP2(K)=SQRT(ABS(RCP2(K)**2-RBC2**2))
      IF(K.NE.3) GO TO 359
      IF(K.EQ.3) THBUV(K)=ATAN(WCP1(K)/ZCP1(K))
      ANG1=ATAN(UCP(K)/VCP(K))
```

```
      IF(UCP(K).LE.0.0) ANG1=ATAN(ABS(UCP(K))/VCP(K))
      ANG2=ATAN(UCP(K)/(C-VCP(K)))
      IF(UCP(K).LE.0.0) ANG2=ATAN(ABS(UCP(K))/(C-VCP(K)))
      AN12=ANG1+ANG2
      ANB1=ATAN(RCCP1(K)/RBC1)
      AN1B=ANB1+ANG1
      IF (UCP(K).GT.0.0) AN1B=ANB1-ANG1
      AN11=PHI-ANG1
      AN22=ANG2+PHI
      IF (UCP(K).GE.0.0) AN11=PHI+ANG1
      IF (UCP(K).GE.0.0) AN22=PHI-ANG2
      RCCN1=SQRT(ABS(RBC1**2+RCP1(K)**2-2.*RBC1*RCP1(K)*COS(AN11)))
      RCCN2=SQRT(ABS(RBC2**2+RCP2(K)**2-2.*RBC2*RCP2(K)*COS(AN22)))
      PPND=RBC1/COS(AN1B)
      RPMOTN=PPND*RPMIN/(C-PPND)
      VELR(I)=RPMIN/RPMOTN
      VEL1=RCP1(K)*RPMIN*2.*PI/12.
      VEL2=RCP2(K)*RPMOTN*2.*PI/12.
      PPP=PPND-RPC1
      VEL11=VEL1
      VEL22=RCP2(K)*RPMOUT*2.*PI/12.
      SV12=SQRT(ABS(VEL1**2+VEL2**2-2.*VEL1*VEL2*COS(AN12)))
      SVS(I)=SV12
      SV11=SQRT(ABS(VEL11**2+VEL22**2-2.*VEL11*VEL22*COS(AN12)))
      OMEGA1=2.*PI*RPMIN/60.
      OMEGA2=2.*PI*RPMOUT/60.
      SLIDV1=SQRT(ABS(RCP1(K)**2-RBC1**2))
      SLIDV2=SQRT(ABS(RCP2(K)**2-RBC2**2))
      SV13=(ABS(OMEGA2*SLIDV2-OMEGA1*SLIDV1))*5.
      SVR=SVS(I)/SV13
      RVELR=RPMOTN/RPMOUT
C INTERFERENCE CHECK
      LINF=1
      NNNN=1
      IF(RCP1(K).LT.RBC1) LINF=2
      IF(RCP2(K).LT.RBC2) LINF=2
      IF(RCP1(K).LT.RBI1) LINF=2
      IF(RCP2(K).LT.RBI2) LINF=2
      IF(VCP(K).GT.0.0) UVDCP=SQRT((UCP(K)-U1(JCP))**2+(VCP(K)-V1(JCP))*
     1*2)
      IF(VCP(K).EQ.0.0) UVDCP=-1.0
      IF(VCP(K).LT.V1(JCP)) UVDCP=-UVDCP
      PHIOP(I)=(ANB1+ANG1)*57.29578
      IF (UCP(K).GT.0.0) PHIOP(I)=(ANB1-ANG1)*57.29578
  359 CONTINUE
  360 CONTINUE
      UU1MAX=0.0
      KKMAX=0
      KKMM=0
      NMAX=0
      IF(RCP1(K).EQ.0.0.OR.RCP2(K).EQ.0.0) GO TO 23
      IF((PSI1TP(K).GT.PSI1LS).OR.(PSI1TP(K).LT.PSI1LE)) GO TO 23
      XCP1(K) = WCP1(K)
      XCP2(K) = WCP2(K)
      YCP1(K) = ZCP1(K) - RR01
```

```
      YCP2(K)=ZCP2(K)-RRO2
   23 CONTINUE
C
      Q1 = 1.0
      KT = 0.0
      PSIR1T = 0.0
      PSIR2T = 0.0
      DO 35 K = 1,5
      TDCP1(K) = 0.0
      TDCP2(K) = 0.0
      THCP1(K) = 0.0
      THCP2(K) = 0.0
      TD(K) = 0.0
      PSIRD1(K) = 0.0
      PSIRD2(K) = 0.0
      H1(K) = 0.0
      H2(K) = 0.0
      HD(K) = 0.0
      CDEFL(K) = 0.0
      STIFF(K) = 0.0
      KT=0.0
      TPS(K,I)=0.0
      MM=7
      IF(RCP1(K).EQ.0.0.OR.RCP2(K).EQ.0.0) MM=35
      IF((PSI1TP(K).GT.PSI1LS).OR.(PSI1TP(K).LT.PSI1LE)) MM=36
      IF(RCP1(K).EQ.0.0.OR.RCP2(K).EQ.0.0) GO TO 35
      IF((PSI1TP(K).GT.PSI1LS).OR.(PSI1TP(K).LT.PSI1LE)) GO TO 35
      DO 24 K1 = 1,L1
      IF(Y1(K1).EQ.YCP1(K).OR.Y1(1).LT.YCP1(K)) GO TO 30
      IF(Y1(K1).LT.YCP1(K)) GO TO 31
   24 CONTINUE
   25 DO 26 K2 = 1,L2
      IF(Y2(K2).EQ.YCP2(K)) GO TO 32
      IF(Y2(K2).LT.YCP2(K)) GO TO 33
   26 CONTINUE
   30 TDCP1(K) = DCP1(K1)
      THCP1(K) = THETC1(K1)
      GO TO 25
   31 CONTINUE
      YINCRM=(Y1(K1-1)-YCP1(K))/(Y1(K1-1)-Y1(K1))
      TDCP1(K)=DCP1(K1-1)+YINCRM*(DCP1(K1)-DCP1(K1-1))
      THCP1(K)=THETC1(K1-1)+YINCRM*(THETC1(K1)-THETC1(K1-1))
      GO TO 25
   32 TDCP2(K) = DCP2(K2)
      THCP2(K) = THETC2(K2)
      GO TO 35
   33 CONTINUE
      YINCRM=(Y2(K2-1)-YCP2(K))/(Y2(K2-1)-Y2(K2))
      TDCP2(K)=DCP2(K2-1)+YINCRM*(DCP2(K2)-DCP2(K2-1))
      THCP2(K)=THETC2(K2-1)+YINCRM*(THETC2(K2)-THETC2(K2-1))
   35 CONTINUE
      DO 36 K = 1,5
      IF(RCP1(K).EQ.0.0.OR.RCP2(K).EQ.0.0) GO TO 36
      IF((PSI1TP(K).GT.PSI1LS).OR.(PSI1TP(K).LT.PSI1LE)) GO TO 36
      TD(K) = TDCP1(K) + TDCP2(K)
```

```
      C1 = (4.*RCCP1(K)*RCCP2(K))/((PI*F)*(RCCP1(K)+RCCP2(K)))
      BH = SQRT(C1*C2*Q1)
      H1(K) = XCP1(K)/COS(THCP1(K))
      H2(K) = XCP2(K)/COS(THCP2(K))
      ARG3 = (2.*H1(K))/BH
      ARG4 = (2.*H2(K))/BH
      HD(K)=(C3*(ALOG(ARG3)-C4)+C5*(ALOG(ARG4)-C6))*Q1
      CDEFL(K) = TD(K) + HD(K)
      STIFF(K) = 1.0/CDEFL(K)
      KT = KT + STIFF(K)
      TPS(K,I)=STIFF(K)
   36 CONTINUE
      CMS(I)=KT
      DO 37 K = 1,5
      QTP(K) = 0.0
      IF (KT.NE.0.0) QTP(K)=(STIFF(K)/KT)*P
      IF (PSI1TP(K).GT.PSI1LS.OR.PSI1TP(K).LT.PSI1LE) GO TO 37
      V11 = QTP(K)*COS(THCP1(K))
      T1 = V11*(YCP1(K) + RRO1)
      PSIRD1(K) = (T1/(4.*PI*F1*G1))*(1./(RI1**2)-1./(RRO1**2))
      PSIR1T = PSIR1T + PSIRD1(K)
      V22 = QTP(K)*COS(THCP2(K))
      T2 = V22*(YCP2(K) + RRO2)
      PSIRD2(K) = (T2/(4.*PI*F2*G2))*(1./(RI2**2)-1./(RRO2**2))
      PSIR2T = PSIR2T + PSIRD2(K)
   37 CONTINUE
      KT=0.0
      DO 40 K = 1,5
      IF(RCP1(K).EQ.0.0.OR.RCP2(K).EQ.0.0) GO TO 40
      IF((PSI1TP(K).GT.PSI1LS).OR.(PSI1TP(K).LT.PSI1LE)) GO TO 40
      THETAQ = ATAN(XCP1(K)/(RRO1 + YCP1(K)))
      ARG = ABS(THCP1(K) + THETAQ)
      STIFF(K) = 1.0/CDEFL(K)
      KT = KT + STIFF(K)
      TPS(K,I)=STIFF(K)
      TDCP1(K) = TDCP1(K)*QTP(K) + RCP1(K)*(PSIR1T-PSIRD1(K))*COS(ARG)
      THETAQ = ATAN(XCP2(K)/(RRO2+YCP2(K)))
      ARG = ABS(THCP2(K) + THETAQ)
      TDCP2(K) = TDCP2(K)*QTP(K) + RCP2(K)*(PSIR2T-PSIRD2(K))*COS(ARG)
      TD(K) = TDCP1(K) + TDCP2(K)
      BH = SQRT(C1*C2*QTP(K))
      ARG3 = (2.*H1(K))/BH
      ARG4 = (2.*H2(K))/BH
      HD(K) = (C3*(ALOG(ARG3)-C4) +  C5*(ALOG(ARG4)-C6))*QTP(K)
      CMS(I)=KT
      CDEFL(K) = TD(K) + HD(K)
   40 CONTINUE
      RNCP(I) = NCTP
      DO 4000 K=2,5
      IF (PSI1TP(K).GT.PSI1LS.OR.PSI1TP(K).LT.PSI1LE) GO TO 4000
      IF (QTP(K).NE.0.0) GO TO 4000
      RCP11=SQRT(UC1(K)**2 + VC1(K)**2)
      RCP22=SQRT(UC2(K)**2 + (C-VC2(K))**2)
      TDCP1(K)=RCP11*(PSIR1T-PSIRD1(K))
      TDCP2(K)=RCP22*(PSIR2T-PSIRD2(K))
```

```
4000 CONTINUE
     DO 43 K = 1,5
     TPS(K,I)=STIFF(K)
     Q(K,I) = QTP(K)
     YC1(K,I) = YCP1(K)
     XC1(K,I) = XCP1(K)
     YC2(K,I) = YCP2(K)
     XC2(K,I) = XCP2(K)
     RC1(K,I) = RCP1(K)
     RC2(K,I) = RCP2(K)
     RCC1(K,I) = RCCP1(K)
     RCC2(K,I)=RCCP2(K)
     TDEFL1(K,I)=TDCP1(K) + HD(K)*DELTA1/(DELTA1+DELTA2)
     TDEFL2(K,I)=TDCP2(K) + HD(K)*DELTA2/(DELTA1+DELTA2)
     IF(K.NE.3) GO TO 41
     RESIDL(1,I)=TDEFL1(3,I)
     RESIDL(2,I)=TDEFL2(3,I)
     IF((VCP(3).NE.0.0).AND.(UCP(3).LE.0.0)) THUVP(I)=ATAN(UCP(3)/VCP(3
    1))-THBUV(I)+THPP1
     IF((VCP(3).NE.0.0).AND.(UCP(3).GT.0.0)) THUVP(I)=ATAN(UCP(3)/VCP(3
    1))-THBUV(I)+THPP1
     IF(VCP(3).EQ.0.0) THUVP(I)=0.0
     THUVP(I)=THUVP(I)*57.29578
  41 CONTINUE
     IF (IPICK.EQ.1HN.OR.QTP(K).LT.5.) GO TO 43
     WRITE(LU,411) K,STIFF(K),QTP(K),TDEFL1(K,I),TDEFL2(K,I),
    &              IPT1(K),PSI1TP(K),IPT2(K),PSI2TP(K)
411  FORMAT(I6,E13.7,F10.2,2E11.5,5X,2(I6,F11.7,3X))
  43 CONTINUE
     TDEF1(I) = TDCP1(3)
     TDEF2(I) = TDCP2(3)
     HDEF(I) = HD(3)
     CDEF(I) = CDEFL(3)
     TPS(3,I)=STIFF(3)
     IF (Q(4,I).GT.1.0.AND.PSI1EP.EQ.0) PSI1EP=PSI1TP(3)
     IF (Q(4,I).GT.1.0.AND.PSI2EP.EQ.0) PSI2EP=PSI2TP(3)
     GO TO 45
  44 TPS(3,I) = 0.0
  45 PSI1TP(1) = PSI1L - FLOAT(I)*DELTA1
     PSI2TP(1) = PSI2L + FLOAT(I)*DELTA2
     HZP=0.0
     IF (RC1(3,I).EQ.0.0.OR.RC2(3,I).EQ.0.0) GO TO 4500
     SRCN=1./RCCN1+1./RCCN2
     HZN=0.564*SQRT((Q(3,I)*SRCN)/(F*C2))
     SHZN=HZN*SVS(I)*0.2
     SRCC=1./RCC1(3,I)+1./RCC2(3,I)
     HZP=0.564*SQRT((Q(3,I)*SRCC)/(F*C2))
     SHZI=HZP*SV13*0.2
     IF (I1.EQ.2) SHZI=HZP*SV12*0.2
     RBCN=RBC1*VELR(I)
     RCC2(3,I)=SQRT(RC2(3,I)**2-RBCN**2)
4500 CONTINUE
  46 CONTINUE
     WRITE(LU,9989)
9989 FORMAT('COMPLETED PART2')
```

```
      IF (I1.EQ.2) GO TO 9990
      ANG(1)=PSI1SL
      ANG(2)=PSI1EP
      ANG(3)=PSI2SL
      ANG(4)=PSI2EP
      GO TO 9991
 9990 ANG(5)=PSI1SL
      ANG(6)=PSI1EP
      ANG(7)=PSI2SL
      ANG(8)=PSI2EP
 9991 GO TO INSTN
      END
C
      PROGRAM PART3,5
C
      COMMON/C7/BUF7(708)
C
      DIMENSION IDCB(144),INAME(3)
      DATA INAME/2HEX,2HDA,2HTA/,ICODE/1/
C
      CALL OPEN(IDCB,IER,INAME,1,-99,36)
      IF (IER.GT.0) GO TO 13
C
9     WRITE(1,10) IER,ICODE
10    FORMAT('FMP ERROR :',I5,' AT ICODE=',I2,'        PROGRAM ABORTED')
      GO TO 50
C
13    ICODE=ICODE+1
      CALL WRITF(IDCB,IER,BUF7,1416)
      IF (IER.LT.0) GO TO 9
C
      WRITE(1,30) INAME
30    FORMAT('DATA FROM SEGMT WAS WRITTEN TO FILE ',3A2)
C
50    CALL CLOSE(IDCB,IER)
      END
C
C
      BLOCK DATA
      COMMON/C1/BUF1(1200)
      COMMON/C2/BUF2(16)
      COMMON/C3/BUF3(17)
      COMMON/C4/BUF4(400)
      COMMON/C5/BUF5(506)
      COMMON/C6/L1,L2,I1,IR1S,IR2E,INSTN
      COMMON/C7/BUF7(708)
C
      END
```

```
C            <830610.1607>
FTN4,L
      PROGRAM EPCYC
C
C*****THIS PROGRAM ASSEMBLES THE SUN/PLANET AND PLANET/RING STIFFNESS FUNCT.
C
      COMMON/C8/RNCPE(50),PSI1E(50),PSI2E(50),QE(5,50),TPSE(5,50),
     &CMSE(50),ANGE(8)
      COMMON/C9/RNCPI(50),PSI1I(50),PSI2I(50),QI(5,50),TPSI(5,50),
     &CMSI(50),ANGI(8)
      COMMON/C10/ PSIS(50,3),KGE(50,3),PSIP(50,3),KGI(50,3),
     &            REPE(3),REPI(3),DUMMY(10)
      COMMON/C11/VRATEX(50,3),VRATIN(50,3)
      COMMON/C12/TPE(50,3),TPI(50,3)
      DIMENSION ITIME(5),KGDATA(3),KEXDAT(3),KINDAT(3),ITDATA(3)
      DIMENSION ISHARE(3),PSI1EX(50),PSI2EX(50),PSI1IN(50),PSI2IN(50)
      REAL KGE,KGI
      DATA KEXDAT/2HEX,2HDA,2HTA/, KINDAT/2HIN,2HDA,2HTA/
      DATA KGDATA/2HKG,2HDA,2HTA/, ITDATA/2HST,2HDA,2HTA/
      DATA ISHARE/2HSH,2HAR,2HE /
      DATA TGS,TGP,TGR/14.,28.,70./, PI/3.1415927/
C
      WRITE(1,1)
    1 FORMAT('LU FOR OUPUT DEVICE?')
      READ(1,*) LU
      CALL EXEC(11,ITIME)
      WRITE(LU,2) ITIME(5),ITIME(4),ITIME(3),ITIME(2),ITIME(1)
    2 FORMAT(/'EPIC2 EXEC START AT ',I5,4(':',I2)/)
C
      CALL TRANR(KEXDAT,8)
      CALL TRANR(KINDAT,9)
      CALL TRANR(KGDATA,10)
      CALL TRANR(ITDATA,11)
      CALL TRANR(ISHARE,12)
C
      DO 6 I=1,50
      PSI1EX(I)=PSI1E(I) - PI/2.
      PSI2EX(I)=(3.*PI/2.) - PSI2E(I)
      PSI1IN(I)=(2.*PI/360.)*(-PSI1I(I))
      PSI2IN(I)=(2.*PI/360.)*(-PSI2I(I))
    6 CONTINUE
      WRITE(1,11)
   11 FORMAT('LIST EXDATA AND INDATA?   [Y/N] _')
      READ(1,12) IPICK
   12 FORMAT(A1)
      IF (IPICK.EQ.1HN) GO TO 14
      WRITE(LU,13) ANGE,ANGI,
     & (I,PSI1E(I),PSI2E(I),TPSE(2,I),TPSE(3,I),CMSE(I),RNCPE(I),
     &  I,PSI1I(I),PSI2I(I),TPSI(2,I),TPSI(3,I),CMSI(I),RNCPI(I),I=1,50)
   13 FORMAT('ANGE',8F13.7/'ANGI',8F13.7//50(2(I5,2F9.7,3E11.5,F9.6)/))
   14 WRITE(1,15)
   15 FORMAT(/'OPTIONS:   1-KGE   2-GPSE   3-KGI   4-GPSI',
     &'   5-RELIST   6-NO CHANGE'/
     &'ENTER CHANGES [**INCLUDE DATA POINTS FOR OPTIONS 1-4**]  > _')
      READ(1,*) ICHNG,NUM
```

```
      GO TO (160,165,170,175,6,19), ICHNG
  160 WRITE(1,161) CMSE(NUM),RNCPE(NUM)
  161 FORMAT('CURRENT VALUES:',2E15.7,/'ENTER NEW VALUES')
      READ(1,*) CMSE(NUM),RNCPE(NUM)
      GO TO 180
  165 WRITE(1,161) TPSE(2,NUM),TPSE(3,NUM)
      READ(1,*) TPSE(2,NUM),TPSE(3,NUM)
      GO TO 180
  170 WRITE(1,161) CMSI(NUM),RNCPI(NUM)
      READ(1,*) CMSI(NUM),RNCPI(NUM)
      GO TO 180
  175 WRITE(1,161) TPSI(2,NUM),TPSI(3,NUM)
      READ(1,*) TPSI(2,NUM),TPSI(3,NUM)
  180 WRITE(1,181) NUM,TPSE(2,NUM),TPSE(3,NUM),CMSE(NUM),RNCPE(NUM),
     &                 TPSI(2,NUM),TPSI(3,NUM),CMSI(NUM),RNCPI(NUM)
  181 FORMAT('NEW VALUES @',I4,2(3F10.1,F10.7,4X))
      GO TO 14
C
   19 DO 20 I=1,50
      IF (QE(4,I).GT.1.0) GO TO 21
   20 CONTINUE
   21 IEPE=I
      ITPE=50-IEPE
      DO 22 I=1,50
      IF (QI(4,I).GT.1.0) GO TO 23
   22 CONTINUE
   23 IEPI=I
      ITPI=50-IEPI
      WRITE(1,24) IEPE,IEPI
   24 FORMAT('IEPE,IEPI',2I5,'   CHANGE?  [Y/N]')
      READ(1,12) IPICK
      IF (IPICK.EQ.1HN) GO TO 28
      WRITE(1,25)
   25 FORMAT('ENTER NEW VALUES FOR IEPE AND IEPI')
      READ(1,*) IEPE,IEPI
      WRITE(1,26) IEPE,IEPI
   26 FORMAT('NEW VALUES FOR IEPE &IEPI',2I6)
      WRITE(LU,27) IEPE,IEPI
   27 FORMAT('IEPE,IEPI',2I5)
C
   28 PRS =ANGI(1)
      PRE =ANGI(2)
      PRSP=ANGI(5)
      PREP=ANGI(6)
      PR  =2*PI - ANGE(3)
      RATE=(PRSP-PREP)/(PRS-PRE)
      PRP =PREP + RATE*(PR-PRE)
      PRPP=PRP - PI/2.
C
      DO 30 I=1,50
      IF (PRPP.GT.PSI1IN(I)) GO TO 31
   30 CONTINUE
   31 HALF=(PSI1IN(I-1) - PSI1IN(I))*0.5 + PSI1IN(I)
      IF (PRPP.GT.HALF) I=I-1
C
```

```
      WRITE(1,32) PRS,PRE,PRSP,PREP,PR,RATE,PRP,PRPP,I
   32 FORMAT('PRS, PRE, PRSP, PREP, PR, RATE, PRP, PRPP, I'/
     &       8F11.7,I5)
C
   33 WRITE(1,34)
   34 FORMAT('WHICH SUN/PLANET/RING TO BE CONSIDERED?
     &       '[HIT "4" TO SKIP]')
      READ(1,*) ISPR
      IF (ISPR.GT.3) GO TO 36
      REPE(ISPR)=FLOAT(IEPE)
      REPI(ISPR)=FLOAT(IEPI)
C
      DO 35 J=1,50
      PSIS(J,ISPR)=PSI1EX(1) - PSI1EX(J)
      PSIP(J,ISPR)=PSI1IN(1) - PSI1IN(J)
      K=J
      IF (K.GT.IEPE-1) K=K-IEPE+1
      KGE(J,ISPR)=CMSE(K)
      IF (CMSE(K).LT.2.5E5) KGE(J,ISPR)=5.E5
      L=I+J-1
      IF (L.GT.IEPI-1) L=L-IEPI+1
      KGI(J,ISPR)=CMSI(L)
      IF (CMSI(L).LT.2.5E5) KGI(J,ISPR)=5.E5
      VRATEX(J,ISPR)=RHCPE(J)
      VRATIN(J,ISPR)=RHCPI(L)
   35 CONTINUE
C
   36 NEWTH=0
      DO 37 J=1,50
      TPE(J,2)=0.
      TPE(J,3)=TPSE(3,J)
      TPI(J,2)=0.0
      L=I+J-1
      IF (L.GT.IEPI-1) NEWTH=1
      IF (L.GT.IEPI-1) L=L-IEPI+1
      TPI(J,3)=TPSI(3,L)
      IF (NEWTH.EQ.1) TPI(J,3)=TPSI(2,L)
   37 CONTINUE
C
      WRITE(1,40)
   40 FORMAT('ANY FURTHER REVISIONS?     [Y/N] _')
      READ(1,12) IPICK
      IF (IPICK.EQ.1HY) GO TO 33
C
   42 WRITE(1,43)
   43 FORMAT('LIST DATA FILE KGDATA?     [Y/N] _')
      READ(1,12) IPICK
      IF (IPICK.EQ.1HN) GO TO 46
      WRITE(1,44)
   44 FORMAT('LU NUMBER OF OUTPUT DEVICE?')
      READ(1,*) LU
      WRITE(LU,45) IEPE,IEPI,(J,(PSIS(J,K),KGE(J,K),K=1,3),
     &             J,(PSIP(J,L),KGI(J,L),L=1,3),J=1,50)
   45 FORMAT('IEPE,IEPI',2I5/50(I2,3(F10.6,F10.1),I6,3(F10.6,F10.1)/))
C
```

```
   46 WRITE(1,47)
   47 FORMAT('LIST DATA FILE STDATA?    [Y/N] _')
      READ(1,12) IPICK
      IF (IPICK.EQ.1HN) GO TO 50
      WRITE(1,44)
      READ(1,*) LU
      WRITE(LU,49) (J,(VRATEX(J,K),K=1,3),J,(VRATIN(J,L),L=1,3),J=1,50)
   49 FORMAT(50(2(I3,3F10.4,5X)/))
C
   50 WRITE(1,51)
   51 FORMAT('LIST DATA FILE SHARE?     [Y/N] _')
      READ(1,12) IPICK
      IF (IPICK.EQ.1HN) GO TO 56
      WRITE(1,44)
      READ(1,*) LU
      WRITE(LU,55) (I,TPE(I,2),TPE(I,3),KGE(I,1),
     &              I,TPI(I,2),TPI(I,3),KGI(I,1),I=1,50)
   55 FORMAT(50(I5,3F10.1,I4,3F10.1/))
   56 WRITE(1,57)
   57 FORMAT('WRITE TO FILE KEXDAT?     [Y/N] _')
      READ(1,12) IPICK
      IF (IPICK.EQ.1HY) CALL TRANW(KEXDAT,8)
      WRITE(1,58)
   58 FORMAT('WRITE TO FILE KINDAT?     [Y/N] _')
      READ(1,12) IPICK
      IF (IPICK.EQ.1HY) CALL TRANW(KINDAT,9)
      WRITE(1,59)
   59 FORMAT('WRITE TO FILE KGDATA?     [Y/N] _')
      READ(1,12) IPICK
      IF (IPICK.EQ.1HY) CALL TRANW(KGDATA,10)
      WRITE(1,61)
   61 FORMAT('WRITE TO FILE ITDATA?     [Y/N] _')
      READ(1,12) IPICK
      IF (IPICK.EQ.1HY) CALL TRANW(ITDATA,11)
      WRITE(1,65)
   65 FORMAT('WRITE TO FILE SHARE?      [Y/N] _')
      READ(1,12) IPICK
      IF (IPICK.EQ.1HY) CALL TRANW(ISHARE,12)
C
      END
```

```
C
C
C
      SUBROUTINE TRANR(INAME,IB)
      COMMON/C8/BUF8(708)
      COMMON/C9/BUF9(708)
      COMMON/C10/BUF10(616)
      COMMON/C11/BUF11(300)
      COMMON/C12/BUF12(300)
C
      DIMENSION IDCB(144),INAME(3)
      DATA ICODE/1/
C
      CALL OPEN(IDCB,IER,INAME,0,-99,36)
      IF (IER.GT.0) GO TO 13
C
9     WRITE(1,10) IER,ICODE,INAME
10    FORMAT('FMP ERROR :',I5,' AT ICODE=',I2,' ON FILE ',3A2/
     &       'PROGRAM ABORTED')
      CALL EXEC(6)
C
13    ICODE=ICODE+1
      IF (IB.EQ.8) CALL READF(IDCB,IER,BUF8)
      IF (IB.EQ.9) CALL READF(IDCB,IER,BUF9)
      IF (IB.EQ.10) CALL READF(IDCB,IER,BUF10)
      IF (IB.EQ.11) CALL READF(IDCB,IER,BUF11)
      IF (IB.EQ.12) CALL READF(IDCB,IER,BUF12)
      IF (IER.LT.0) GO TO 9
C
      WRITE(1,30) INAME
30    FORMAT('DATA FROM ',3A2,' WAS READ')
50    CALL CLOSE(IDCB,IER)
      RETURN
      END
```

```
C
C
      SUBROUTINE TRANW(INAME,IB)
      COMMON/C8/BUF8(708)
      COMMON/C9/BUF9(708)
      COMMON/C10/BUF10(616)
      COMMON/C11/BUF11(300)
      COMMON/C12/BUF12(300)
      DIMENSION IDCB(144),INAME(3)
      DATA ICODE/1/
C
      CALL OPEN(IDCB,IER,INAME,0,-99,36)
      IF (IER.GT.0) GO TO 13
C
9     WRITE(1,10) IER,ICODE,INAME
10    FORMAT('FMP ERROR :',I5,' AT ICODE=',I2,' ON FILE ',3A2/
     &       'PROGRAM ABORTED')
      CALL EXEC(6)
C
13    ICODE=ICODE+1
      IF (IB.EQ.8) CALL WRITF(IDCB,IER,BUF8,1416)
      IF (IB.EQ.9) CALL WRITF(IDCB,IER,BUF9,1416)
      IF (IB.EQ.10) CALL WRITF(IDCB,IER,BUF10,1212)
      IF (IB.EQ.11) CALL WRITF(IDCB,IER,BUF11,600)
      IF (IB.EQ.12) CALL WRITF(IDCB,IER,BUF12,600)
      IF (IER.LT.0) GO TO 9
C
      WRITE(1,30) INAME
30    FORMAT('DATA FROM EPIC2 WAS WRITTEN TO FILE ',3A2)
50    CALL CLOSE(IDCB,IER)
      RETURN
      END
C
C
C
      BLOCK DATA
      COMMON/C8/BUF8(708)
      COMMON/C9/BUF9(708)
      COMMON/C10/BUF10(616)
      COMMON/C11/BUF11(300)
      COMMON/C12/BUF12(300)
C
      END
```

```
C                    <830611.1140>
FTN4,L
      PROGRAM INTEG
C
C*****THIS PROGRAM INTEGRATES DIFFERENTIAL EQUATIONS OF MOTION FOR A NINE
C*****D.O.F. PGT USING A 4th ORDER RUNGE KUTTA TECHNIQUE
C
      COMMON/C10/PSIE(50,3),KGE(50,3),PSII(50,3),KGI(50,3),REPE(3),
     &           REPI(3),RBCS,RBCP,RBCR,TOUT,ANGLE(6)
      COMMON/C11/VRE(50,3),VRI(50,3)
      COMMON/C13/XI(18),DXI(18),NE,NP,NRK,IP,LPP,ISTORE,JEP,NCT
      COMMON/C14/TIME(400),RESULT(15,400),ENDPT
C
      DIMENSION IEPE(3),PSIEPE(3),IEPI(3),PSIEPI(3),PHI(3),ALPHA(3)
      DIMENSION PSPP(3),PSPPD(3),PSPPDD(3),KGPE(3),KGPI(3),TRE(3),TRI(3)
      DIMENSION CGPE(3),CGPI(3),GPTFE(3),GPTFI(3),ITIME(5),DFE(3),DFI(3)
      DIMENSION PSIAE(3),PSIAI(3),RBCPP(3),RBCRP(3),KDAT(3),IDAT(3)
      DIMENSION CRME(3),CRMI(3),INAME(3)
      REAL M,JD,JS,JP,JC,JR,JL,KDS,KLS,KG,LDS,LLS,KEQ,KSUN,MSUN
      REAL KGE,KGI,KGPE,KGPI
      DATA PI/3.1415927/, INAME/2HTR,2HAN,2HW /
      DATA N1,N2/1,2/, KDAT/2HKG,2HDA,2HTA/, IDAT/2HST,2HDA,2HTA/
C*****
      DATA DP,PHID/5.0000000,22.5/, DELS,DELR/2*.01/
      DATA ES,EP,ER/3*30.E6/, PRS,PRP,PRR/3*.285/
      DATA GAMAS,GAMAP,GAMAR/3*.288/, ZETAK/.050/
      DATA FWS,FWP,FWR/3*1.0/, RD,RL/1.5,1.5/, ZETAS,ZETAG/.005,.1000/
      DATA TGS,TGP,TGR/25.,24.,73./, TIN/4500.0/
C
      WRITE(1,1)
    1 FORMAT('ENTER LU FOR OUTPUT DEVICE')
      READ(1,*) LU
      WRITE(1,3)
    3 FORMAT('PRINT INTERMEDIATE RESULTS? [Y/N]')
      READ(1,4) IWRIT
    4 FORMAT(A1)
      CALL EXEC(11,ITIME)
      WRITE(LU,2) ITIME(5),ITIME(4),ITIME(3),ITIME(2),ITIME(1)
    2 FORMAT('FINAB EXEC START AT',I5,4(':',I2))
C
      CALL TRANR(KDAT,10)
      CALL TRANR(IDAT,11)
C
    6 PHIR=PHID*PI/180.
      PDS=TGS/DP
      PDP=TGP/DP
      PDR=TGR/DP
      RPCS=0.5*PDS
      RPCP=0.5*PDP
      RPCR=0.5*PDR
      RC=RPCS+RPCP
      CP=PI/DP
      BP=CP*COS(PHIR)
      RBCS=RPCS*COS(PHIR)
      RBCP=RPCP*COS(PHIR)
```

```
      RBCR=RPCR*COS(PHIR)
      PITCHE=2.*PI/TGS
      PITCHI=2.*PI/TGP
      DO 7 I=1,3
      IEPE(I)  =IFIX(REPE(I))
      IEPI(I)  =IFIX(REPI(I))
      PSIEPE(I)=PSIE(IEPE(I))
      PSIEPI(I)=PSII(IEPI(I))
      PHI(I)   =PSIEPE(I)*(I-1)/3.
      ALPHA(I)=PI - PHIR - (I-1)*2.*PI/3.
    7 CONTINUE
      WRITE(1,10)
   10 FORMAT('ENTER VALUE FOR KSUN, MSUN, CSUN AND INPUT RPM')
      READ(1,*) KSUN,MSUN,CSUN,RPMIN
C
C     IF(JS.EQ.0.) JS=.5*GAMAS*PI*FWS*RPCS**4/386.
C     IF(JP.EQ.0.) JP=.5*GAMAP*PI*FWP*RPCP**4/386.
C     IF(JR.EQ.0.) JR=3.*JP + JP*(386./((RPCP**4)*(RPCR**2)))
C     IF(JD.EQ.0.0) JD = 50.*JR
C     IF(JL.EQ.0.0) JL = 50.*JR
C     IF(LDS.EQ.0.0) LDS = 6.
C     IF(LLS.EQ.0.0) LLS = 6.
C     GDS = 30000000./(2.*(1. + .285))
C     GLS = 30000000./(2.*(1. + .285))
C     RL  = RD * SQRT(TGR/TGS)
C     KDS = (PI*((2.*RD)**4)*GDS)/(32.*LDS)
C     KLS = (PI*((2.*RL)**4)*GLS)/(32.*LLS)
C     CDS = (2.*ZETAS*SQRT(KDS))/SQRT((JD+JS)/(JD*JS))
C     CLS = (2.*ZETAS*SQRT(KLS))/SQRT((JL+JR)/(JL*JR))
C     CSUN=  2.*ZETAK*SQRT(KSUN*MSUN)
C     MSUN=JS/(RPBS**2)
C
      JD   = 5.
      JS   =  .04
      MSUN =  .015
      JP   =  .033
      JR   = 2.
      JL   = 5.
      KDS  = 8.85E5
      KLS  = 8.85E5
      CDS  = 0.65
      CLS  = 3.5
      CSUN =  .04
C
C
C*****SET UP EQUIVALENT GEAR SYSTEM
C
      RVALU  = TGR/(TGR+TGS)
      RPMIN  = RVALU*RPMIN
      RPMOUT = (TGS/TGR)*RPMIN
      TOUT   = (TGR/TGS)*TIN
C
      DOMGAD = 2.*PI*(RPMIN/60.)
      DOMGAS = DOMGAD
      DOMGAP = (TGS/TGP)*DOMGAS
```

```
      DOMGAR = (TGS/TGR)*DOMGAS
      DOMGAL = DOMGAR
C
      CYCLE  = 2.*PI/TGS
      PERIOD = CYCLE/DOMGAD
      DT     = 0.0
C
      WRITE(1,20) PERIOD
   20 FORMAT('PERIOD OF PITCH CYCLE:',F11.7,' SECONDS'/
     &       'ENTER NUMBER OF INTEGRATION CYCLES AND STEPS PER CYCLE')
      READ(1,*) NCTR,STEP
      DDT=PERIOD/STEP
      WRITE(1,21) DDT
   21 FORMAT('TIME STEP ',F10.7/'ENTER ADDITIONAL PITCH',
     &           ' CYCLES AND HOW OFTEN TO SAVE RESULTS')
      READ(1,*) LSAVE,JSAVE
      NCT=NCTR+LSAVE
      WRITE(LU,22) NCT,PERIOD,DDT,JSAVE
   22 FORMAT('INTEG CYCLES:',I4,' PERIOD:',F9.7,' TIME STEP:',F9.7,
     &       ' EVERY',I4'th CYCLE SAVED')
C
      CALL POSIT(LU,KSUN,ALPHA,TGS,TGR)
C
      DPSID = TIN/KDS
      DPSIS = 0.0
      DPSIR = ANGLE(1)
      DPSI1 = ANGLE(2)
      DPSI2 = ANGLE(3)
      DPSI3 = ANGLE(4)
      DPSIL = DPSIR - TOUT/KLS
C
      DPSIDD = DOMGAD
      DPSISD = DOMGAS
      DPSI1D = DOMGAP
      DPSI2D = DOMGAP
      DPSI3D = DOMGAP
      DPSIRD = DOMGAR
      DPSILD = DOMGAL
C
      WRITE(LU,5006) TGS,TGP,TGR,DP,PHID,
     &               RPCS,RPCP,RPCR,RC,RBCS,RBCP,RBCR
 5006 FORMAT('TGS,TGP,TGR      ',3F5.1,' Pd',F7.3,' PHI',F5.1/
     &       'RPCS,RPCP,RPCR,RC',4F14.7/'RBCS,RBCP,RBCR   ',4F14.7)
      WRITE(LU,5001) JD,JS,MSUN,JP,JR,JL, KDS,KLS,KSUN, CDS,CLS,CSUN,
     &      PSIEPE,PSIEPI, IEPE,IEPI, RPMIN,RPMOUT,TIN,TOUT
 5001 FORMAT('JD,JS,MS,JP,JR,JL',6F10.6/
     &       'KDS,KLS,KSUN     ',3F11.1/
     &       'CDS,CLS,CSUN     ',3F11.7/
     &       'PSIEPE,PSIEPI    ',6F10.7/
     &       'IEPE,IEPI      ',3I5,5X,3I5/
     &       'RPMI,RPMO,TI,TO  ',5F10.3)
      WRITE(LU,5002) DPSID,DPSIS,DPSI1,DPSI2,DPSI3,DPSIR,DPSIL,
     &               DPSIDD,DPSISD,DPSI1D,DPSI2D,DPSI3D,DPSIRD,DPSILD
 5002 FORMAT(/'INITIAL POSITIONS AND VELOCITIES: DRIVER, SUN, PLANETS',
     &       ' (1,2,&3), RING, AND LOAD'/7F11.7/7F11.5//)
```

```
C
C
      PSDP = DPSID
      PSSP = DPSIS
      XS   = ANGLE(5)
      YS   = ANGLE(6)
      PSPP(1) = DPSI1
      PSPP(2) = DPSI2
      PSPP(3) = DPSI3
      PSRP = DPSIR
      PSLP = DPSIL
C
      PSDPD = 0.
      PSSPD = 0.
      PSPPD(1) = 0.
      PSPPD(2) = 0.
      PSPPD(3) = 0.
      PSRPD = 0.
      PSLPD = 0.
C
C
C
      NP = 0
      NRK = 1
      J = 0
      LC = 0
      LP = 0
C     TSP = 0.0
C     TEP = TTOTAL
C     PRI = DDT
C     PT = TSP
      IP = 0
      LPP = 0
C     ILP = 0
      ISTORE = 0
C     DELTAT = DDT
      SSTEP=0.
      JWRIT=0
      NTH=0
      NWRIT=0
      ISAVE=JSAVE
      INDEX=1
      EXTRA=PITCHE/STEP
C
C
C
  275 ARGE=DPSIS/PITCHE
      NC=IFIX(ARGE) + 1
      IF (NC.GT.NCT) GO TO 5060
      NC=IFIX(ARGE + EXTRA) + 1
C
      DO 295 J=1,3
      PSIAE(J)=(ARGE-FLOAT(IFIX(ARGE)))*PSIEPE(J) + PHI(J)
      IF (PSIAE(J).GT.PSIEPE(J)) PSIAE(J)=PSIAE(J) - PSIEPE(J)
C
```

```
      DO 280 I=2,IEPE(J)
      IF (PSIAE(J).LE.PSIE(I,J)) GO TO 285
  280 CONTINUE
  285 KGPE(J)=KGE(I-1,J) + (KGE(I,J)-KGE(I-1,J))*(PSIAE(J)-PSIE(I-1,J))/
     &                                            (PSIE(I,J)-PSIE(I-1,J))
      IF (PSIAE(J).GT.PSIE(IEPE(J),J)) KGPE(J)=KGE(IEPE(J)+1,J)
      IF (KGPE(J).LT.0.) WRITE(1,286) J,PSIAE(J),KGPE(J)
  286 FORMAT('EX>0.',I3,F10.7,F10.1)
      CGPE(J)=(2.*ZETAG*SQRT(KGPE(J)))/SQRT((RBCS**2)/JS+(RBCP**2)/JP)
      TRE(J)=VRE(I-1,J) + (VRE(I,J)-VRE(I-1,J))*(PSIAE(J)-PSIE(I-1,J))/
     &                                            (PSIE(I,J)-PSIE(I-1,J))
      IF (PSIAE(J).GT.PSIE(IEPE(J),J)) TRE(J)=VRE(IEPE(J)+1,J)
C
      PSIAI(J)=PSIAE(J)*TGS/TGP
      IF (PSIAI(J).GT.PSIEPI(J)) PSIAI(J)=PSIAI(J) - PSIEPI(J)
C
      DO 290 I=2,IEPI(J)
      IF (PSIAI(J).LE.PSII(I,J)) GO TO 291
  290 CONTINUE
  291 KGPI(J)=KGI(I-1,J) + (KGI(I,J)-KGI(I-1,J))*(PSIAI(J)-PSII(I-1,J))/
     &                                            (PSII(I,J)-PSII(I-1,J))
      IF (PSIAI(J).GT.PSII(IEPI(J),J)) KGPI(J)=KGI(IEPI(J)+1,J)
      IF (KGPI(J).LT 0.) WRITE(1,292) J,PSIAE(J),PSIAI(J),KGPI(J)
  292 FORMAT('IN. < 0.',I3,2F10.7,F10.1)
      CGPI(J)=(2.*ZETAG*SQRT(KGPI(J)))/SQRT((RBCS**2)/JP+(RBCR**2)/JP)
      TRI(J)=VRI(I-1,J) + (VRI(I,J)-VRI(I-1,J))*(PSIAI(J)-PSII(I-1,J))/
     &                                            (PSII(I,J)-PSII(I-1,J))
      IF (PSIAI(J).GT.PSII(IEPI(J),J)) TRI(J)=VRI(IEPI(J)+1,J)
C     TRE(J)=TGP/TGS
C     TRI(J)=TGR/TGP
  295 CONTINUE
  296 CONTINUE
C
C
C*****GEAR PAIR TRANSMITTED FORCE, EXTERNAL AND INTERNAL*****************
C
      CMFE=0.
      CMFI=0.
      FX=0.
      FY=0.
      TOUT  =0.
      TORQUE=0.
      DO 300 I=1,3
      RBCPP(I)=RBCS*TRE(I)
      CRME(I) =RBCS*PSSP - RBCPP(I)*PSPP(I)
      IF (CRME(I).LT.0.0) GO TO 302
  301 GPTFE(I)=CGPE(I)*(RBCS*(PSSPD+DOMGAS) -
     &                 (RBCPP(I)*PSPPD(I) + RBCP*DOMGAP) +
     &          XSD*COS(ALPHA(I)) + YSD*SIN(ALPHA(I))) +
     &          KGPE(I)*(RBCS*(PSSP+DT*DOMGAS) -
     &                 (RBCPP(I)*PSPP(I) + RBCP*DT*DOMGAP) +
     &          XS*COS(ALPHA(I)) + YS*SIN(ALPHA(I)))
      GO TO 305
  302 IF (ABS(CRME(I)).GT.DELS) GO TO 303
      GPTFE(I)=0.0
```

```
      GO TO 305
  303 GPTFE(I)=CGPE(I)*(RBCS*(PSSFD+DOMGAS) -
     &                  (RBCPP(I)*PSPPD(I) + RBCP*DOMGAP) +
     &          XSD*COS(ALPHA(I)) + YSD*SIN(ALPHA(I))) +
     &          KGPE(I)*(RBCS*(PSSP+DT*DOMGAS) -
     &                  (RBCPP(I)*PSPP(I) + RBCP*DT*DOMGAP) + DELS
     &          XS*COS(ALPHA(I)) + YS*SIN(ALPHA(I)))
  305 CMFE=CMFE+GPTFE(I)
      FX=FX + GPTFE(I)*COS(ALPHA(I) - PI)
      FY=FY + GPTFE(I)*SIN(ALPHA(I) - PI)
C
      RBCRP(I)=RBCP*TRI(I)
      CRMI(I) =RBCP*PSPP(I) - RBCRP(I)*PSRP
      IF (CRMI(I).LT.0.0) GO TO 307
  306 GPTFI(I)=CGPI(I)*(RBCRP(I)*PSRPD + RBCR*DOMGAR -
     &                  RBCP*(PSPPD(I)+DOMGAP)) +
     &          KGPI(I)*(RBCRP(I)*PSRP + RBCR*DT*DOMGAR -
     &                  RBCP*(PSPP(I)+DT*DOMGAP))
      GO TO 310
  307 IF (ABS(CRMI(I)).GT.DELR) GO TO 308
      GPTFI(I)=0.0
      GO TO 310
  308 GPTFI(I)=CGPI(I)*(RBCRP(I)*PSRPD + RBCR*DOMGAR -
     &                  RBCP*(PSPPD(I)+DOMGAP)) +
     &          KGPI(I)*(RBCRP(I)*PSRP + RBCR*DT*DOMGAR -
     &                  RBCP*(PSPP(I)+DT*DOMGAP) - DELR)
  310 CMFI=CMFI+GPTFI(I)
      TOUT  =TOUT + TIN*TRE(I)*TRI(I)/3.
      TORQUE=TORQUE + GPTFI(I)*RBCRP(I)
  300 CONTINUE
C
C*****THE EQUATIONS OF MOTION*************************************************
C
C
      PSDPDD=(-CDS*((PSDPD+DOMGAD) - (PSSPD+DOMGAS))
     &        -KDS*((PSDP+DT*DOMGAD) - (PSSP+DT*DOMGAS)) + TIN)/JD
C
C
      PSSPDD=(-CDS*((PSSPD+DOMGAS) - (PSDPD+DOMGAD))
     &        -KDS*((PSSP+DT*DOMGAS) - (PSDP+DT*DOMGAD))
     &        -(RBCS*CMFE))/JS
C
      XSDD=(-CSUN*XSD -KSUN*XS +FX)/MSUN
C
      YSDD=(-CSUN*YSD -KSUN*YS +FY)/MSUN
C
C
      DO 312 I=1,3
      PSPPDD(I)=(RBCPP(I)*GPTFE(I) + RBCP*GPTFI(I))/JP
  312 CONTINUE
C
C
      PSRPDD=(-CLS*((PSRPD+DOMGAR) - (PSLPD+DOMGAL))
     &        -KLS*((PSRP+DT*DOMGAR) - (PSLP+DT*DOMGAL))
     &        -TORQUE)/JR
```

```
C
C
      PSLPDD=(-CLS*((PSLPD+DOMGAL) - (PSRPD+DOMGAR))
     &        -KLS*((PSLP+DT*DOMGAL) - (PSRP+DT*DOMGAR)) -TOUT)/JL
C
      IF(NRK.EQ.1) GO TO 320
C
  315 CONTINUE
C     WRITE(LU,5010)     DT,PSDPDD,PSSPDD,XSDD,YSDD,(PSPPDD(I),I=1,3),
C    & PSRPDD,PSLPDD,PSDPD,PSSPD,XSD,YSD,(PSPPD(I),I=1,3),PSRPD,PSLPD,
C    &                        PSDP,PSSP,XS,YS,(PSPP(I),I=1,3),PSRP,PSLP
 5010 FORMAT(F9.7,2F13.7,2E15.7,5F13.7/9X,2F13.7,2E15.7,5F13.7/
     &         9X,2F13.7,2E15.7,5F13.7)
C
      CALL RKUTA(DT,DDT)
      CALL MOREK(PSDP,PSDPD,DDT)
      CALL MOREK(PSDPD,PSDPDD,DDT)
      CALL MOREK(PSSP,PSSPD,DDT)
      CALL MOREK(PSSPD,PSSPDD,DDT)
      CALL MOREK(XS,XSD,DDT)
      CALL MOREK(XSD,XSDD,DDT)
      CALL MOREK(YS,YSD,DDT)
      CALL MOREK(YSD,YSDD,DDT)
      DO 316 I=1,3
      CALL MOREK(PSPP(I),PSPPD(I),DDT)
      CALL MOREK(PSPPD(I),PSPPDD(I),DDT)
  316 CONTINUE
      CALL MOREK(PSRP,PSRPD,DDT)
      CALL MOREK(PSRPD,PSRPDD,DDT)
      CALL MOREK(PSLP,PSLPD,DDT)
      CALL MOREK(PSLPD,PSLPDD,DDT)
C
C     WRITE(LU,5010)     DT,PSDPDD,PSSPDD,XSDD,YSDD,(PSPPDD(I),I=1,3),
C    & PSRPDD,PSLPDD,PSDPD,PSSPD,XSD,YSD,(PSPPD(I),I=1,3),PSRPD,PSLPD,
C    &                        PSDP,PSSP,XS,YS,(PSPP(I),I=1,3),PSRP,PSLP
C
      DPSIS = PSSP + DT*DOMGAS
      GO TO 275
C
320   DPSID = PSDP + DT*DOMGAD
      DPSI1 = PSPP(1) + DT*DOMGAP
      DPSI2 = PSPP(2) + DT*DOMGAP
      DPSI3 = PSPP(3) + DT*DOMGAP
      DPSIR = PSRP + DT*DOMGAR
      DPSIL = PSLP + DT*DOMGAL
C
      DPSIDD = PSDPD + DOMGAD
      DPSISD = PSSPD + DOMGAS
      DPSI1D = PSPPD(1) + DOMGAP
      DPSI2D = PSPPD(2) + DOMGAP
      DPSI3D = PSPPD(3) + DOMGAP
      DPSIRD = PSRPD + DOMGAR
      DPSILD = PSLPD + DOMGAL
C
      T = DT
```

```
      SSTEP=SSTEP+1.
C
      EFTORQ=0.
      DO 5020 I=1,3
      DFE(I)=GPTFE(I)
      DFI(I)=GPTFI(I)
      EFTORQ=EFTORQ + DFI(I)*RBCP*TRI(I)
 5020 CONTINUE
C
      IF (IWRIT.EQ.1HY) WRITE(LU,5007) T,(DFE(I),I=1,3),(DFI(I),I=1,3),
     &                                 XS,YS
 5007 FORMAT(F9.7,2X,3F7.1,2X,3F7.1,2X,2F10.7)
C
      JTH=IFIX(NC/2.)
      IF (JTH.LT.JWRIT) GO TO 6004
      CALL EXEC(11,ITIME)
      WRITE(1,6003) NC,ITIME(4),ITIME(3),ITIME(2)
 6003 FORMAT('START ',I4,'th INTEGRATION CYCLE @ ',I2,2(':',I2))
      JWRIT=JWRIT+1
 6004 IF (NC.LE.NCTR) GO TO 315
C
      IF (ISAVE.NE.JSAVE) GO TO 5009
      ISAVE=0
      TIME(INDEX)=T*DONGAS/PITCHE
      RESULT(1,INDEX)=KGPE(1)
      RESULT(2,INDEX)=KGPE(2)
      RESULT(3,INDEX)=KGPE(3)
      RESULT(4,INDEX)=DFE(1)
      RESULT(5,INDEX)=DFE(2)
      RESULT(6,INDEX)=DFE(3)
      RESULT(7,INDEX)=KGPI(1)
      RESULT(8,INDEX)=KGPI(2)
      RESULT(9,INDEX)=KGPI(3)
      RESULT(10,INDEX)=ABS(DFI(1))
      RESULT(11,INDEX)=ABS(DFI(2))
      RESULT(12,INDEX)=ABS(DFI(3))
      RESULT(13,INDEX)=XS
      RESULT(14,INDEX)=YS
      RESULT(15,INDEX)=EFTORQ
      INDEX=INDEX+1
      IF (INDEX.GT.400) GO TO 5058
 5009 ISAVE=ISAVE+1
C
      NTH=NTH+1
      IF (NTH.NE.1) GO TO 5026
      CALL EXEC(11,ITIME)
      WRITE(1,5025) NCTR,ITIME(5),ITIME(4),ITIME(3),ITIME(2),ITIME(1)
 5025 FORMAT(/'CONCLUDE',I4,' CYLES OF INTEGRATION AT',I5,4(':',I2)//)
      WRITE(1,1)
      READ(1,*) LU
      WRITE(LU,5025) NCTR,ITIME(5),ITIME(4),ITIME(3),ITIME(2),ITIME(1)
      GO TO 5027
C
 5026 ITH=IFIX(NTH/50.)
      IF (ITH.LT.NWRIT) GO TO 315
```

```
 5027 IF (LU.EQ.1) WRITE(LU,5007) T,(DFE(I),I=1,3),(DFI(I),I=1,3),XS,YS
      IF (LU.EQ.6) WRITE(LU,5029) T,ARGE,(DFE(I),I=1,3),(DFI(I),I=1,3),
     &            XS,YS
 5029 FORMAT(F9.7,F13.5,3X,3F8.1,3X,3F8.1,3X,2F12.7)
 5050 NWRIT=NWRIT+1
C
      GO TO 315
 5058 WRITE(LU,5059)
 5059 FORMAT('*****INDEX IS GREATER THAN 400,INTEGRATION IS CONCLUDED')
 5060 CONTINUE
C
      INDEX=INDEX-1
 6000 WRITE(1,6001) ARGE,SSTEP,INDEX
 6001 FORMAT(F9.2,' PITCH CYCLES COMPLETED',F8.1,' INTEGRATION CYCLES'/
     &I4,' DATA VALUES SAVED, WRITE RESULTS TO DATA FILE?  [Y/N]   _')
      READ(1,4) IWRIT
      IF (IWRIT.NE.1HY.AND.IWRIT.NE.1HN) GO TO 6000
      ENDPT=INDEX*1.
      IF (IWRIT.EQ.1HY) CALL TRANW
      IF (IWRIT.EQ.1HN) WRITE(LU,6002)
 6002 FORMAT('RESULTS NOT WRITTEN TO DATA FILE')
C
      END
```

```
C
      SUBROUTINE POSIT(LU,KSUN,ALPHA,TGS,TGR)
      COMMON/C10/PSIE(50,3),KGE(50,3),PSII(50,3),KGI(50,3),REPE(3),
     &           REPI(3),RBCS,RBCP,RBCR,TOUT,C(6)
      COMMON/C11/TRE(50,3),TRI(50,3)
      DIMENSION A(6,6),B(6),ALPHA(3),FE(3),FI(3)
****************!!DO NOT FORGET THESE PHASE ANGLES
      REAL KGE,KGI,KSUN
      INTEGER PHIE(3),PHII(3)
      DATA PHIE/1,12,23/, PHII/1,10,19/
C
      NPLAN=3
      N=6
      DO 1 I=1,N
      DO 1 J=1,N
      A(I,J)=0.
      B(I)=0.
      C(I)=0.
    1 CONTINUE
C     DO 5 I=1,3
C     TRE(PHIE(I),I)=2.0
C     TRI(PHII(I),I)=2.5
C   5 CONTINUE
C
C*****ASSEMBLE LOWER-TRIANGULAR STIFFNESS MATRIX AND TORQUE VECTOR
C
C     SIGMA=0.
      A(1,1)=0.
      SUMXX=0.
      SUMXY=0.
      SUMYY=0.
      ER   =0.
      DO 2 I=1,NPLAN
C     SIGMA=SIGMA+KGI(PHII(I),I)
      A(1,1)=A(1,1) + ((RBCP*TRI(PHII(I),I))**2)*KGI(PHII(I),I)
      SUMXX=SUMXX + KGE(PHIE(I),I)*(COS(ALPHA(I))**2)
      SUMXY=SUMXY + KGE(PHIE(I),I)*COS(ALPHA(I))*SIN(ALPHA(I))
      SUMYY=SUMYY + KGE(PHIE(I),I)*(SIN(ALPHA(I))**2)
      ER=ER + TRE(PHIE(I),I)*TRI(PHII(I),I)
C
      J=I+1
C     A(J,1)=-RBCR*RBCP*KGI(PHII(I),I)
C     A(J,J)= RBCP*RBCP*(KGE(PHIE(I),I)+KGI(PHII(I),I))
C     A(5,J)=-RBCP*KGE(PHIE(I),I)*COS(ALPHA(I))
C     A(6,J)=-RBCP*KGE(PHIE(I),I)*SIN(ALPHA(I))
C
      A(J,1)=-(RBCP*TRI(PHII(I),I)) * RBCP * KGI(PHII(I),I)
      A(J,J)=((RBCS*TRE(PHIE(I),I))**2)*KGE(PHIE(I),I) +
     &       (RBCP**2)*KGI(PHII(I),I)
      A(5,J)=-(RBCS*TRE(PHIE(I),I))*KGE(PHIE(I),I)*COS(ALPHA(I))
      A(6,J)=-(RBCS*TRE(PHIE(I),I))*KGE(PHIE(I),I)*SIN(ALPHA(I))
    2 CONTINUE
C
C     A(1,1)=RBCR*RBCR*SIGMA
      A(5,5)=KSUN + SUMXX
```

```
      A(6,5)=SUMXY
      A(6,6)=KSUN + SUMYY
      B(1)=-TOUT*(ER/(NPLAN*TGR/TGS))
C
C*****FILL SYMMETRIC NxN STIFFNESS MATRIX
C
      DO 10 I=1,N-1
      DO 10 J=I+1,N
      A(I,J)=A(J,I)
   10 CONTINUE
C
C     WRITE(LU,11) ((A(I,J),J=1,N),B(I),I=1,N)
   11 FORMAT(/6(F9.1,5F12.1,F10.1/))
C
C*****PERFORM GAUSS ELIMINATION
C
      DO 40 I=1,N-1
      DO 30 J=I+1,N
      DIV=-A(J,I)/A(I,I)
      DO 20 K=I,N
      A(J,K)=A(J,K)+DIV*A(I,K)
   20 CONTINUE
      B(J)=B(J)+DIV*B(I)
   30 CONTINUE
C     WRITE(LU,11) ((A(L,M),M=1,N),B(L),L=1,N)
   40 CONTINUE
C
C*****PERFORM BACK-SUBSTITUTION
C
      C(N)=B(N)/A(N,N)
      DO 60 II=1,N-1
      I=N-II
      SUM=B(I)
      DO 50 JJ=1,N-1
      J=N+1-JJ
      SUM=SUM-C(J)*A(I,J)
   50 CONTINUE
      C(I)=SUM/A(I,I)
   60 CONTINUE
C
      TI=0.
      TO=0.
      DO 65 I=1,3
      FI(I)=(RBCP*TRI(PHII(I),I)*C(1) - RBCP*C(I+1)) * KGI(PHII(I),I)
      FE(I)=(RBCS*TRE(PHIE(I),I)*C(I+1) - C(5)*COS(ALPHA(I)) -
     &                                C(6)*SIN(ALPHA(I))) * KGE(PHIE(I),I)
      TI=TI - RBCS*FE(I)
      TO=TO + RBCP*TRI(PHII(I),I)*FI(I)
   65 CONTINUE
      WRITE(LU,70) (TRE(PHIE(I),I),TRI(PHII(I),I),I=1,3), (C(I),I=1,N)
   70 FORMAT('EXTERNAL, INTERNAL TRANSMISSION RATIOS'/3(2F10.6,5X)/
     &    'RING, PLANET ROTATIONS, AND SUN CENTER DISPLACEMENT'/6E13.6)
      WRITE(LU,75) FE,FI,TI,TO
   75 FORMAT('EX MESH FORCES',3F8.2,2X,'IN MESH FORCES',3F8.2/
     &       'INPUT TORQUE',F9.2,10X,'EFFECTIVE OUPUT TORQUE',F9.2/)
```

```
C
      RETURN
      END
C
C
C
      SUBROUTINE RKUTA(T,DT)
      COMMON/C13/XI(18),DXI(18),NE,NP,NRK,IP,LPP,ISTORE,JEP,NCT
      NE = 0
      NP = NP + 1
      NRK = 0
      IF(NP.EQ.5) NP = 1
      IF(NP.EQ.1) GO TO 1
      IF(NP.EQ.2) RETURN
      IF(NP.EQ.3) GO TO 2
      IF(NP.EQ.4) NRK = 1
      RETURN
    1 DT = DT/2.
      T = T + DT
      RETURN
    2 T = T + DT
      DT = 2.*DT
      RETURN
      END
C
C
      SUBROUTINE MOREK(X,DX,DT)
      COMMON/C13/XI(18),DXI(18),NE,NP,NRK,IP,LPP,ISTORE,JEP,NCT
      NE = NE + 1
      GO TO (1,2,3,4),NP
    1 XI(NE) = X
      DXI(NE) = DX
      X = X + DX*DT
      RETURN
    2 DXI(NE) = DXI(NE) + 2.*DX
      X = XI(NE) + DX*DT
      RETURN
    3 DXI(NE) = DXI(NE) + 2.*DX
      X = XI(NE) + DX*DT
      RETURN
    4 DXI(NE) = (DXI(NE) + DX)/6.
      X = XI(NE) + DXI(NE)*DT
      RETURN
      END
```

```
C
C
C
      SUBROUTINE TRANR(INAME,IB)
      COMMON/C10/BUF10(616)
      COMMON/C11/BUF11(300)
      DIMENSION IDCB(144),INAME(3)
      DATA ICODE/1/
C
      CALL OPEN(IDCB,IER,INAME,1,-99,36)
      IF (IER.GT.0) GO TO 13
9     WRITE(1,10) IER,ICODE,INAME
10    FORMAT('FMP ERROR :',I5,' AT ICODE=',I2,' ON FILE ',3A2/
     &       'PROGRAM ABORTED')
      CALL CLOSE(IDCB,IER)
      CALL EXEC(6)
13    ICODE=ICODE+1
      IF (IB.EQ.10) CALL READF(IDCB,IER,BUF10)
      IF (IB.EQ.11) CALL READF(IDCB,IER,BUF11)
      IF (IER.LT.0) GO TO 9
      ICODE=ICODE+1
      WRITE(1,30) INAME
30    FORMAT('DATA FROM ',3A2,' WAS READ')
50    CALL CLOSE(IDCB,IER)
      RETURN
      END
C
C
C
      SUBROUTINE TRANW
      COMMON/C14/TIME(400),RESULT(15,400),ENDPT
      DIMENSION IDCB(144),INAME(3),ITITLE(64),BUF(15)
C
      ICODE=1
      N=IFIX(ENDPT)
      WRITE(1,10)
   10 FORMAT('ENTER DATA FILE NAME')
      READ(1,15) INAME
   15 FORMAT(64A2)
      WRITE(1,20)
   20 FORMAT('ENTER HEADING LESS THAN 129 CHARACTERS')
      READ(1,15) ITITLE
      WRITE(1,15) ITITLE
      CALL OPEN(IDCB,IER,INAME,1,-99,36)
      IF (IER.GT.0) GO TO 30
   25 WRITE(1,26) IER,ICODE
   26 FORMAT('FMP ERROR',I4,' AT ICODE=',I4/'PROGRAM ABENDED')
      ICODE=ICODE+1
   30 CALL WRITF(IDCB,IER,ITITLE,64)
      IF (IER.LT.0) GO TO 25
      ICODE=ICODE+1
      CALL WRITF(IDCB,IER,ENDPT,2)
      IF (IER.LT.0) GO TO 25
      DO 60 I=1,N
```

```
      BUF(1)=TIME(I)
      DO 40 J=2,16
      BUF(J)=RESULT(J-1,I)
   40 CONTINUE
      ICODE=ICODE+1
      CALL WRITF(IDCB,IER,BUF,32)
      IF (IER.LT.0) GO TO 25
   60 CONTINUE
      DUMMY=999.
      ICODE=ICODE+1
      CALL WRITF(IDCB,IER,DUMMY,-1)
      IF (IER.LT.0) GO TO 25
      WRITE(1,70) INAME
   70 FORMAT('RESULTS FROM FINAB SUCCESSFULLY WRITTEN TO DATA FILE ',
     &        3A2)
      CALL CLOSE(IDCB,IER)
      STOP
      END
C
C
C
      BLOCK DATA
      COMMON/C10/BUF10(616)
      COMMON/C11/BUF11(300)
      COMMON/C13/BUF13(36),IBUF13(8)
      COMMON/C14/BUF14(6401)
C
      END
```

```
C     <830611.1003>
FTN4,L
      PROGRAM FORCE
C
C*****THIS PROGRAM PERFORMS THE STATIC ANALYSIS FOR A PGT
C*****INCLUDES STATIC MESH AND TOOTH LOADS AS WELL AS SUN GEAR DISPLACEMENT
C
      COMMON/C10/PSIE(50,3),KGE(50,3),PSII(50,3),KGI(50,3),REPE(3),
     &           REPI(3),RBCS,RBCP,RBCR,TOUT,C(6)
      COMMON/C11/VRE(50,3),VRI(50,3)
      COMMON/C12/TPSE(50,3),TPSI(50,3)
C
      DIMENSION ITIME(5),KGPE(3),KGPI(3),IEPE(3),IEPI(3),PSIEPE(3)
      DIMENSION IGCB(192),PSIEPI(3),PHI(3),PSIAE(3),PSIAI(3),ALPHA(3)
      DIMENSION V1(2),V2(2),V3(2),V4(2),XTIC(2),YTIC(2),XMAJ(2),YMAJ(2),
     &YLAST(2),NPOINT(2),TRE(3),TRI(3),KDAT(3),IDAT(3),PSI(200),
     &VAL(2,200),ISHR(3),FE2(200),FE3(200),FI2(200),FI3(200)
      REAL KGE,KGI,KGPE,KGPI,KSUN
      DATA PI/3.1415927/, KDAT/2HKG,2HDA,2HTA/, IDAT/2HST,2HDA,2HTA/
      DATA ISHR/2HSH,2HAR,2HE /
C
      DATA DP,PHID/5.000,22.5/
      DATA TGS,TGP,TGR/25.,24.,73./, RPMIN,TIN/1000.,4500.0/
C
      DATA NPOINT/1,5/, XTIC(2),YTIC(1),XMAJ(2),YMAJ(1),YLAST/6*0./
C
      WRITE(1,11)
   11 FORMAT('ENTER LU FOR OUTPUT AND ID,LU FOR GRAPHICS DEVICE'/
     &'1-CRT  2-PRT;          1,1-CRT  2,57-4PP')
      READ(1,*) IU,ID,LU
    3 FORMAT(A1)
      WRITE(1,4)
    4 FORMAT('WHICH PLOT?  1-X,Y vs PSI  2-X vs Y  3-MESH FORCES')
      READ(1,*) NTH
C
      CALL TRANR(KDAT,10)
      CALL TRANR(IDAT,11)
      CALL TRANR(ISHR,12)
      IF (ID.EQ.1) CALL SCREN(V1,V2,V3,V4)
      IF (ID.EQ.2) CALL FOUR(V1,V2,V3,V4)
C
    6 PHIR=PHID*PI/180.
      PDS=TGS/DP
      PDP=TGP/DP
      PDR=TGR/DP
      RPCS=0.5*PDS
      RPCP=0.5*PDP
      RPCR=0.5*PDR
      TOUT=TIN*(TGR/TGS)
      RBCS=RPCS*COS(PHIR)
      RBCP=RPCP*COS(PHIR)
      RBCR=RPCR*COS(PHIR)
      DO 7 I=1,3
      IEPE(I)  =IFIX(REPE(I))
      IEPI(I)  =IFIX(REPI(I))
```

```
      PSIEPE(I)=PSIE(IEPE(I))
      PSIEPI(I)=PSII(IEPI(I))
      PHI(I)   =PSIEPE(I) * (I-1)/3.
      ALPHA(I)=PI - PHIR - (I-1)*2.*PI/3.
    7 CONTINUE
      PITCHE=2.*PI/TGS
      PITCHI=2.*PI/TGP
C
      WRITE(1,8) TGS,TGP,TGR,TIN,IEPE,IEPI
    8 FORMAT('TGS,TGP,TGR,TIN,IEPE1,IEPI1',2X,3F6.1,F8.1,2(2X,3I3))
      WRITE(1,9)
    9 FORMAT('ENTER SUN SUPPORT STIFFNESS, START CYCLE, NO. CYCLES, ',
     &       'INTERVALS PER CYCLE'/'[4 VALUES REQD]  > _')
      READ(1,*) KSUN,ARG,NCT,DIV
      DPSIS=PITCHE*ARG
      NCT=NCT+IFIX(ARG)
C
      W1=ARG
      W2=FLOAT(NCT)
      WRITE(1,273)
  273 FORMAT('ENTER W3, W4 AND YTIC SPACING')
      READ(1,*) W3,W4,YTIC(2)
      XTIC(1)=-.1
      XMAJ(1)=10.
      YMAJ(2)=2.
C
      CALL PLOTR(IGCB,ID,1,LU)
C     IF (ID.EQ.2) CALL LIMIT(IGCB,0.,193.5,5.5,261.5)
      CALL CSIZE(IGCB,2.1,.7,0.)
      CALL PEN(IGCB,1)
      IF (NTH.EQ.2) GO TO 110
C
      DO 105 I=1,1
      CALL VIEWP(IGCB,V1(I),V2(I),V3(I),V4(I))
      CALL WINDW(IGCB,W1,W2,W3,W4)
      DO 105 J=1,2
      CALL FXD(IGCB,NPOINT(J))
      CALL LAXES(IGCB,XTIC(J),YTIC(J),W1,W3,XMAJ(J),YMAJ(J),.5)
  105 CONTINUE
      GO TO 120
  110 CONTINUE
      TC=(W4-W3)/4.
      IF (ID.EQ.1) CALL VIEWP(IGCB,58.,142.,14.,98.)
      IF (ID.EQ.2) CALL VIEWP(IGCB,25.,85.,35.,95.)
      CALL WINDW(IGCB,W3,W4,W3,W4)
      CALL FXD(IGCB,4)
      CALL LGRID(IGCB,-TC,TC,W3,W3,2.,2.)
C
  120 CONTINUE
      DELTA=PSIEPE(1)/DIV
      ITOTAL=IFIX(DIV)+1
      NTIME=0
      ICTR=1
      XOLD=W1
C
```

```
  275 ARGE=DPSIS/PITCHE
C
      DO 295 J=1,3
      PSIAE(J)=(ARGE-FLOAT(IFIX(ARGE)))*PSIEPE(J) + PHI(J)
      IF (PSIAE(J).GT.PSIEPE(J)) PSIAE(J)=PSIAE(J) - PSIEPE(J)
C
      DO 280 I=2,IEPE(J)
      IF (PSIAE(J).LE.PSIE(I,J)) GO TO 285
  280 CONTINUE
  285 EINCRM=(PSIAE(J)-PSIE(I-1,J))/(PSIE(I,J)-PSIE(I-1,J))
      KGPE(J)=KGE(I-1,J) + (KGE(I,J)-KGE(I-1,J))*EINCRM
      IF (PSIAE(J).GT.PSIE(IEPE(J),J)) KGPE(J)=KGE(IEPE(J)+1,J)
      TRE(J) =VRE(I-1,J) + (VRE(I,J)-VRE(I-1,J))*EINCRM
      IF (PSIAE(J).GT.PSIE(IEPE(J),J)) TRE(J)=VRE(IEPE(J)+1,J)
      IF (J.NE.1) GO TO 286
      GPSE3 =TPSE(I-1,3) + (TPSE(I,3)-TPSE(I-1,3))*EINCRM
      IF (PSIAE(J).GT.PSIE(IEPE(J),J)) GPSE3=TPSE(IEPE(J)+1,3)
C
C
  286 PSIAI(J)=PSIAE(J)*TGS/TGP
      IF (PSIAI(J).GT.PSIEPI(J)) PSIAI(J)=PSIAI(J) - PSIEPI(J)
C
      DO 290 I=2,IEPI(J)
      IF (PSIAI(J).LE.PSII(I,J)) GO TO 291
  290 CONTINUE
  291 DINCRM=(PSIAI(J)-PSII(I-1,J))/(PSII(I,J)-PSII(I-1,J))
      KGPI(J)=KGI(I-1,J) + (KGI(I,J)-KGI(I-1,J))*DINCRM
      IF (PSIAI(J).GT.PSII(IEPI(J),J)) KGPI(J)=KGI(IEPI(J)+1,J)
      TRI(J) =VRI(I-1,J) + (VRI(I,J)-VRI(I-1,J))*DINCRM
      IF (PSIAI(J).GT.PSII(IEPI(J),J))  TRI(J)=VRI(IEPI(J)+1,J)
      IF (J.NE.1) GO TO 292
      GPSI3 =TPSI(I-1,3) + (TPSI(I,3)-TPSI(I-1,3))*DINCRM
      IF (PSIAI(J).GT.PSII(IEPI(J),J)) GPSI3=TPSE(IEPI(J)+1,3)
  292 CONTINUE
  295 CONTINUE
C
      CALL POSIT(IU,ALPHA,KSUN,KGPE,KGPI,TRE,TRI,TGS,TGP,TGR)
C
      PSI(ICTR)=ARGE
      IF (NTH.EQ.3) GO TO 296
      VAL(1,ICTR)=C(5)
      VAL(2,ICTR)=C(6)
      GO TO 297
  296 VAL(1,ICTR)=KGPE*ABS(C(2)*RBCP - C(5)*COS(ALPHA(1)) -
     &                                 C(6)*SIN(ALPHA(1)))
      VAL(2,ICTR)=KGPI(1)*(C(2)*RBCP - C(1)*RBCR)
      FE3(ICTR)=(GPSE3/KGPE(1))*VAL(1,ICTR)
      FE2(ICTR)=VAL(1,ICTR)-FE3(ICTR)
C     WRITE(1,2960) PSI(ICTR),GPSE3,KGPE(1),FE2(ICTR),FE3(ICTR),
C    &              VAL(1,ICTR)
      FI3(ICTR)=(GPSI3/KGPI(1))*VAL(2,ICTR)
      FI2(ICTR)=VAL(2,ICTR)-FI3(ICTR)
C     WRITE(1,2960) PSI(ICTR),GPSI3,KGPI(1),FI2(ICTR),FI3(ICTR),
C    &              VAL(2,ICTR)
 2960 FORMAT(F11.7,2F11.1,3F10.2)
```

```
C
  297 DPSIS=DPSIS+DELTA
      IF (IPICK.EQ.1HY) WRITE(IU,298) ARGE,KGPE,KGPI,(VAL(I,ICTR),I=1,2)
  298 FORMAT(F4.2,5X,3F10.1,5X,3F10.1,5X,2F15.7)
      ICTR=ICTR+1
      IF (ICTR.GT.ITOTAL) GO TO 398
      GO TO 275
C
  398 CONTINUE
      IF (NTH.EQ.2) GO TO 410
      FMAX=-1E10
      FMIN=1E10
      DO 409 I=1,1
      CALL VIEWP(IGCB,V1(I),V2(I),V3(I),V4(I))
      CALL WINDW(IGCB,W1,W2,W3,W4)
      CALL LINE(IGCB,0)
      DO 399 J=1,ITOTAL
      IF (J.EQ.1) CALL MOVE(IGCB,PSI(J),VAL(I,J))
      CALL DRAW(IGCB,PSI(J),VAL(I,J))
      IF (VAL(I,J).GT.FMAX) FMAX=VAL(I,J)
      IF (VAL(I,J).LT.FMIN) FMIN=VAL(1,J)
  399 CONTINUE
      IF (NTH.EQ.1.OR.I.EQ.2) GO TO 402
      FEXMAX=-1.E5
      FINMAX=-1.E5
      CALL LINE(IGCB,2)
      DO 400 J=1,ITOTAL
      IF (J.EQ.1) CALL MOVE(IGCB,PSI(J),FE2(J))
      CALL DRAW(IGCB,PSI(J),FE2(J))
  400 CONTINUE
      CALL LINE(IGCB,1)
      DO 401 J=1,ITOTAL
      IF (J.EQ.1) CALL MOVE(IGCB,PSI(J),FE3(J))
C     CALL DRAW(IGCB,PSI(J),FE3(J))
      IF (FE3(J).GT.FEXMAX) FEXMAX=FE3(J)
  401 CONTINUE
  402 IF(NTH.EQ.1.OR.I.EQ.1) GO TO 409
      CALL LINE(IGCB,2)
      DO 406 J=1,ITOTAL
      IF (J.EQ.1) CALL MOVE(IGCB,PSI(J),FI2(J))
C     CALL DRAW(IGCB,PSI(J),FI2(J))
  406 CONTINUE
      CALL LINE(IGCB,1)
      DO 407 J=1,ITOTAL
      IF (J.EQ.1) CALL MOVE(IGCB,PSI(J),FI3(J))
C     CALL DRAW(IGCB,PSI(J),FI3(J))
      IF (FI3(J).GT.FINMAX) FINMAX=FI3(J)
  407 CONTINUE
  409 CONTINUE
      GO TO 430
C
  410 DO 420 I=1,ITOTAL
      IF (I.EQ.1) CALL MOVE(IGCB,VAL(1,I),VAL(2,I))
      CALL DRAW(IGCB,VAL(1,I),VAL(2,I))
  420 CONTINUE
```

```
C
  430 CONTINUE
      WRITE(1,440) FMAX,FMIN,FEXMAX,FINMAX
  440 FORMAT('MAX, MIN MESH LOADS',2F10.2/'FEXMAX,FINMAX',2F10.2)
      CALL PEN(IGCB,0)
      CALL PLOTR(IGCB,ID,0)
      CALL EXEC(6)
      END
C
C
      SUBROUTINE TRANR(INAME,IB)
      COMMON/C10/BUF10(616)
      COMMON/C11/BUF11(300)
      COMMON/C12/BUF12(300)
      DIMENSION IDCB(144),INAME(3)
      DATA ICODE/1/
C
      CALL OPEN(IDCB,IER,INAME,1,-99,36)
      IF (IER.GT.0) GO TO 13
9     WRITE(1,10) IER,ICODE,INAME
10    FORMAT('FMP ERROR :',I5,' AT ICODE=',I2,' ON FILE ',3A2/
     &       'PROGRAM ABORTED')
      CALL CLOSE(IDCB,IER)
      CALL EXEC(6)
13    ICODE=ICODE+1
      IF (IB.EQ.10) CALL READF(IDCB,IER,BUF10)
      IF (IB.EQ.11) CALL READF(IDCB,IER,BUF11)
      IF (IB.EQ.12) CALL READF(IDCB,IER,BUF12)
      IF (IER.LT.0) GO TO 9
      ICODE=ICODE+1
      WRITE(1,30) INAME
30    FORMAT('DATA FROM ',3A2,' WAS READ')
50    CALL CLOSE(IDCB,IER)
      RETURN
      END
```

```
C
C
C
      SUBROUTINE POSIT(LU,ALPHA,KSUN,KE,KI,TRE,TRI,TGS,TGP,TGR)
      COMMON/C10/PSIE(50,3),KGE(50,3),PSII(50,3),KGI(50,3),REPE(3),
     &           REPI(3),RBCS,RBCP,RBCR,TOUT,C(6)
      DIMENSION A(6,6),B(6),ALPHA(3),TRE(3),TRI(3),FE(3),FI(3)
      DIMENSION KE(3),KI(3)
      REAL KSUN,KE,KI,KGE,KGI
C
      NPLAN=3
      N=6
      DO 1 I=1,N
      DO 1 J=1,N
      A(I,J)=0.
      B(I)=0.
      C(I)=0.
    1 CONTINUE
      DO 5 I=1,3
      TRE(I)=TGP/TGS
      TRI(I)=TGR/TGP
    5 CONTINUE
C
C*****ASSEMBLE LOWER-TRIANGULAR STIFFNESS MATRIX AND TORQUE VECTOR
C*****ASSEMBLE LOWER-TRIANGULAR STIFFNESS MATRIX AND TORQUE VECTOR
C
      A(1,1)=0.
      SUMXX=0.
      SUMXY=0.
      SUMYY=0.
      ER   =0.
      DO 2 I=1,NPLAN
      A(1,1)=A(1,1) + ((RBCP*TRI(I))**2)*KI(I)
      SUMXX=SUMXX + KE(I)*(COS(ALPHA(I))**2)
      SUMXY=SUMXY + KE(I)*COS(ALPHA(I))*SIN(ALPHA(I))
      SUMYY=SUMYY + KE(I)*(SIN(ALPHA(I))**2)
      ER=ER + TRE(I)*TRI(I)
C
      J=I+1
      A(J,1)=-(RBCP*TRI(I)) * RBCP * KI(I)
      A(J,J)=((RBCS*TRE(I))**2)*KE(I) +
     &       (RBCP**2)*KI(I)
      A(5,J)=-(RBCS*TRE(I))*KE(I)*COS(ALPHA(I))
      A(6,J)=-(RBCS*TRE(I))*KE(I)*SIN(ALPHA(I))
    2 CONTINUE
C
      A(5,5)=KSUN + SUMXX
      A(6,5)=SUMXY
      A(6,6)=KSUN + SUMYY
      B(1)=-TOUT*(ER/(NPLAN*(TGR/TGS)))
C
C*****FILL SYMMETRIC NxN STIFFNESS MATRIX
C
      DO 10 I=1,N-1
      DO 10 J=I+1,N
```

```
      A(I,J)=A(J,I)
   10 CONTINUE
C     WRITE(LU,11) ((A(I,J),J=1,N),B(I),I=1,N)
   11 FORMAT(/6(F9.1,5F12.1,F10.1/))
C
C*****PERFORM GAUSS ELIMINATION
C
      DO 40 I=1,N-1
      DO 30 J=I+1,N
      DIV=-A(J,I)/A(I,I)
      DO 20 K=I,N
      A(J,K)=A(J,K)+DIV*A(I,K)
   20 CONTINUE
      B(J)=B(J)+DIV*B(I)
   30 CONTINUE
C     WRITE(LU,11) ((A(L,M),M=1,N),B(L),L=1,N)
   40 CONTINUE
C
C*****PERFORM BACK-SUBSTITUTION
C
      C(N)=B(N)/A(N,N)
      DO 60 II=1,N-1
      I=N-II
      SUM=B(I)
      DO 50 JJ=1,N-1
      J=N+1-JJ
      SUM=SUM-C(J)*A(I,J)
   50 CONTINUE
      C(I)=SUM/A(I,I)
   60 CONTINUE
C
C     TI=0.
C     TO=0.
C     DO 65 I=1,3
C     FI(I)=(RBCP*TRI(I)*C(1) - RBCP*C(I+1)) * KI(I)
C     FE(I)=(RBCS*TRE(I)*C(I+1) - C(5)*COS(ALPHA(I)) -
C    &                            C(6)*SIN(ALPHA(I))) * KE(I)
C     TI=TI - RBCS*FE(I)
C     TO=TO + RBCP*TRI(I)*FI(I)
C  65 CONTINUE
C     WRITE(LU,70) (TRE(I),TRI(I),I=1,3), (C(I),I=1,N)
C  70 FORMAT('EXTERNAL, INTERNAL TRANSMISSION RATIOS'/3(2F10.6,5X)/
C    &    'RING, PLANET ROTATIONS, AND SUN CENTER DISPLACEMENT'/6E13.6)
C     WRITE(LU,75) FE,FI,TI,TO
C  75 FORMAT('EX MESH FORCES',3F8.2,2X,'IN MESH FORCES',3F8.2/
C    &       'INPUT TORQUE',F9.2,10X,'EFFECTIVE OUPUT TORQUE',F9.2)
C
      RETURN
      END
```

```
      SUBROUTINE FOUR(BACK)
      DIMENSION BACK(8),VALU(8)
      DATA VALU/2*22.,2*65.,71.,26.,91.,46./
      DO 10 I=1,8
   10 BACK(I)=VALU(I)
      WRITE(1,20)
   20 FORMAT('ENTER V1,V2,V3,V4    > _')
      READ(1,*) BACK(1),BACK(3),BACK(5),BACK(7)
      RETURN
      END
C
C
      SUBROUTINE SCREN(BACK)
      DIMENSION BACK(8),VALU(80)
      DATA VALU/2*20.,2*195.,54.,5.,98.,49./
      DO 10 I=1,8
   10 BACK(I)=VALU(I)
      RETURN
      END
C
C
      BLOCK DATA
      COMMON/C10/BUF10(616)
      COMMON/C11/BUF11(300)
      COMMON/C12/BUF12(300)
C
      END
```

```
C                    <830611.1038>
FTN4,L
      PROGRAM DRAW1
C
C*****THIS PROGRAM EXAMINES AND GRAPHS THE USER SPECIFIED DATA FILES
C*****GENERATED BY PROGRAM INTEG
C
      COMMON/C15/TIME(400),RESULT(3,400)
      DIMENSION IGCB(192),ITITLE(16),INAME(3)
      DATA ID,LU,LS/1,1,0/
      DATA V1,V2,V3,V4/15.,195.,75.,99./
      DATA N1,N2/1,1/, W3,W4/0.,1300./
C
      WRITE(1,100)
  100 FORMAT('ENTER ID,LU FOR PLOTTING DEVICE  [(1,1),(2,57)] > _')
      READ(1,*) ID,LU
      CALL PLOTR(IGCB,ID,1,LU)
C     IF (ID.EQ.2) CALL LIMIT(IGCB,0.,66.,0.,100.)
      WRITE(1,101)
  101 FORMAT('ENTER CHARACTER SIZE   > _')
      READ(1,*) CHAR
      CALL CSIZE(IGCB,CHAR,.75)
C
   10 CALL TRANR(INAME,ICR)
C
      WRITE(1,11)
   11 FORMAT('ENTER FIRST, LAST DATA POINTS, XDIV, YDIV')
      READ(1,*) ISTRT,ISTOP,XDIV,YDIV
      W1=TIME(ISTRT)
      W2=TIME(ISTOP)
C
   12 WRITE(1,1)
    1 FORMAT('ENTER NUMBER OF THE VARIABLE TO BE PLOTTED [1-3]',
     &       '  ENTER 15 FOR XvsY')
      READ(1,*) I
C
      WRITE(1,2) V1,V2,V3,V4,W1,W2,W3,W4
    2 FORMAT('VIEWP:',4F7.1/'WINDW:',4E14.6/
     &       'CHANGE PARAMETERS?  [Y/N]  > _')
      READ(1,3) IPICK
    3 FORMAT(A1)
      IF (IPICK.NE.1HY) GO TO 9
      WRITE(1,4)
    4 FORMAT('CHANGE VIEWPORT?    [Y/N]  > _')
      READ(1,3) IPICK
      IF (IPICK.NE.1HY) GO TO 6
      WRITE(1,5)
    5 FORMAT('ENTER NEW VALUES FOR V1,V2,V3,V4')
      READ(1,*) V1,V2,V3,V4
    6 WRITE(1,7)
    7 FORMAT('CHANGE WINDOW?      [Y/N]  > _')
      READ(1,3) IPICK
      IF (IPICK.NE.1HY) GO TO 9
      WRITE(1,8)
    8 FORMAT('ENTER NEW VALUES FOR W3,W4 (ENTER W1,W2 FIRST IF I=15)')
```

```
      IF (I.NE.15) READ(1,*) W3,W4
      IF (I.EQ.15) READ(1,*) W1,W2,W3,W4
    9 CONTINUE
C
C
      CALL VIEWP(IGCB,V1,V2,V3,V4)
      CALL WINDW(IGCB,W1,W2,W3,W4)
C
      A1=(W2-W1)/XDIV
      A2=(W4-W3)/YDIV
      A3=W1
      A4=W3
      WRITE(1,15) N1,N2
   15 FORMAT('DECIMAL PLACES ON LABELS:',2I4/
     &       'CHANGE?                [Y/N]  > _')
      READ(1,3) IPICK
      IF (IPICK.EQ.1HN) GO TO 17
      WRITE(1,16)
   16 FORMAT('ENTER VALUES FOR DECIMAL PLACES (2 REQD)')
      READ(1,*) N1,N2
   17 CALL FXD(IGCB,N1)
      CALL LAXES(IGCB,A1,0.,A3,A4,1.,0.,.5)
      CALL FXD(IGCB,N2)
      CALL LAXES(IGCB,0.,A2,A3,A4,0.,1.,.5)
C
      IF (I.EQ.15) GO TO 21
      DO 20 J=ISTRT,ISTOP
      IF (J.EQ.ISTRT) CALL MOVE(IGCB,TIME(J),RESULT(I,J))
      CALL DRAW(IGCB,TIME(J),RESULT(I,J))
   20 CONTINUE
      GO TO 23
C
   21 DO 22 J=ISTRT,ISTOP
      IF (J.EQ.ISTRT) CALL MOVE(IGCB,RESULT(1,J),RESULT(2,J))
      CALL DRAW(IGCB,RESULT(1,J),RESULT(2,J))
   22 CONTINUE
   23 CONTINUE
C     WRITE(1,24)
C  24 FORMAT('DIGITIZE?           [Y/N]  > _')
C     READ(1,3) IPICK
C     IF (IPICK.EQ.1HN) GO TO 30
C     WRITE(1,25)
C  25 FORMAT('POSITION CURSOR AND HIT ANY KEY')
C     CALL DIGTZ(IGCB,XPT,YPT)
C     WRITE(1,26) XPT,YPT
C  26 FORMAT(/'COORDINATES OF CURSOR',2F10.6/)
C     GO TO 23
C
   30 WRITE(1,31)
   31 FORMAT('REDRAW?             [Y/N]  > _')
      READ(1,3) IPICK
      IF (IPICK.NE.1HN) GO TO 12
      WRITE(1,32)
   32 FORMAT('PLOT NEW DATA SET?  [Y/N]  > _')
      READ(1,3) IPICK
```

```
      IF (IPICK.NE.1HN) GO TO 10
C
      WRITE(1,39)
   39 FORMAT('LABELS?              [Y/N]  > _')
      READ(1,3) IPICK
      IF (IPICK.NE.1HY) GO TO 60
      CALL VIEWP(IGCB,0.,200.,0.,100.)
      CALL WINDW(IGCB,0.,200.,0.,100.)
C     CALL CSIZE(IGCB,5.0,.75)
C     CALL MOVE(IGCB,70.,56.)
C     CALL LABEL(IGCB)
C     WRITE(LU,41)
C  41 FORMAT('SUN/PLANET MESH FORCES (TYP)')
C     CALL MOVE(IGCB,70.,6.)
C     CALL LABEL(IGCB)
C     WRITE(LU,42)
C  42 FORMAT('PLANET/RING MESH FORCES (TYP)')
      WRITE(1,45)
   45 FORMAT('ENTER GRAPH TITLE  [32 OR LESS CHARACTERS]')
      READ(1,46) ITITLE
   46 FORMAT(16A2)
      CALL MOVE(IGCB,140.,95.)
      CALL LABEL(IGCB)
      WRITE(LU,46) ITITLE
      CALL PEN(IGCB,0)
   60 CALL PLOTR(IGCB,ID,0)
      CALL EXEC(6)
      END

C
C
      SUBROUTINE TRANR(INAME,ICR)
      COMMON/C15/TIME(400),RESULT(3,400)
      DIMENSION IDCB(144),INAME(3),ITITLE(64),BUF(16)
      DATA ITITLE/64*2H  /, ICODE/1/
C
      WRITE(1,5)
    5 FORMAT('READ FROM NEW DATA FILE?    [Y/N] > _')
      READ(1,6) IPICK
    6 FORMAT(A1)
      IF (IPICK.EQ.1HN) GO TO 17
      WRITE(1,10)
   10 FORMAT('ENTER DATA FILE NAME              > _')
      READ(1,15) INAME
   15 FORMAT(64A2)
      WRITE(1,16)
   16 FORMAT('ENTER CARTRIDGE NUMBER            > _')
      READ(1,*) ICR
   17 CALL OPEN(IDCB,IER,INAME,1,-99,ICR)
      IF (IER.GT.0) GO TO 30
   20 WRITE(1,25) IER,ICODE
   25 FORMAT('FMP ERROR',I4,' AT ICODE=',I4/'PROGRAM ABENDED')
      CALL EXEC(6)
   30 ICODE=ICODE+1
```

```
      CALL READF(IDCB,IER,ITITLE)
      IF (IER.LT.0) GO TO 20
      WRITE(1,15) ITITLE
      ICODE=ICODE+1
      CALL READF(IDCB,IER,ENDPT)
      IF (IER.LT.0) GO TO 20
      WRITE(1,40) ENDPT
   40 FORMAT(F6.1,'TIME VALUES STORED'/
     &       'ENTER NUMBER TO ACCESS            > _')
      READ(1,*) ISTOP
      WRITE(1,41)
   41 FORMAT(/'1-EX STIFF  2-EX FORCE  3-IN STIFF  4-IN FORCE',
     &  '  5-X,Y MOVE; TORQUE'/'ENTER DATA SET TO GRAPH          > _')
      READ(1,*) ISET
      DO 60 I=1,ISTOP
      ICODE=ICODE+1
      CALL READF(IDCB,IER,BUF)
      IF (IER.LT.0) GO TO 20
      TIME(I)=BUF(1)
      DO 50 J=1,3
      K=(3*ISET) - 2 + J
      RESULT(J,I)=BUF(K)
   50 CONTINUE
   60 CONTINUE
      CALL CLOSE(IDCB,IER)
      RETURN
      END
C
C
      BLOCK DATA
      COMMON/C15/BUF15(1600)
      END
```

☆U.S. GOVERNMENT PRINTING OFFICE:1984 -739 -010/ 12 REGION NO. 4

1. Report No.	2. Government Accession No.	3. Recipient's Catalog No.
NASA CR-3793		

4. Title and Subtitle	5. Report Date
Dynamics of Planetary Gear Trains	June 1984
	6. Performing Organization Code

7. Author(s)	8. Performing Organization Report No.
R. August, R. Kasuba, J. L. Frater, A. Pintz	None
	10. Work Unit No.

9. Performing Organization Name and Address	11. Contract or Grant No.
Cleveland State University Cleveland, Ohio 44115	NAG3-186
	13. Type of Report and Period Covered
12. Sponsoring Agency Name and Address	Contractor Report
National Aeronautics and Space Administration Washington, D.C. 20546	14. Sponsoring Agency Code 505-40-42 (E-2026)

15. Supplementary Notes

Final report. Project Manager, John J. Coy, Structures and Mechanical Technologies Division, NASA Lewis Research Center, Cleveland, Ohio 44135.

16. Abstract

This investigation has developed a comprehensive method for analyzing the static and dynamic loads in a planetary gear train. A variable-variable mesh stiffness (VVMS) model was used to simulate the external and internal spur gear mesh behavior, and an equivalent conventional gear train concept was adapted for the dynamic studies. No current technique uses a non-linear tooth mesh stiffness in determining component, as well as gear teeth, loading; nor have any investigations been undertaken examining the effect of the phase relationships of the variable mesh stiffness on the dynamic behavior of the gears. With this technique, the design and analysis of planetary gearing is based on an engineering rather than an empirical approach. The analysis is applicable for either involute or non-involute, i.e., modified, spur gearing. By utilizing the equivalent gear train concept, the developed method may be extended for use for all types of epicyclic gearing. This method is incorporated into a computer program so that the static and dynamic behavior of individual components can be examined. In addition to the multi-phased VVMS, other items considered in the analysis are: (1) static and dynamic load sharing among the planets; (2) floating or fixed sun gear; (3) actual tooth geometry, including errors and modifications; (4) positioning errors of the planet gears; (5) torque variations due to non-involute gear action. A mathematical model comprised of power source, load, and planetary transmission is used to determine the instantaneous loads to which the components are subjected. It is capable of considering fluctuating output torque, elastic behavior in the system, and loss of contact between gear teeth. The dynamic model has nine degrees of freedom resulting in a set of simultaneous second-order differential equations with time varying coefficients, which are solved by numerical methods. The computer program was used to conduct parametric studies to determine the effect of manufacturing errors, damping and component stiffness, and transmitted load on dynamic behavior. The results of the analysis indicate this methodology will offer the designer/analyst a comprehensive tool with which planetary drives may be quickly and effectively evaluated.

17. Key Words (Suggested by Author(s))	18. Distribution Statement
Gears Vibrations Planetary gear train Machine design	Unclassified - unlimited STAR Category 37

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of pages	22. Price*
Unclassified	Unclassified	235	A11

*For sale by the National Technical Information Service, Springfield, Virginia 22161