Microgravity Induced Changes in Horizontal Vestibulo-Ocular Reflexes of SLS-1 & SLS-2 Astronauts

by

Matthew A. Neimark

S.B. Massachusetts Institute of Technology, 1995

Submitted to the Department of Electrical Engineering and Computer Science
in Partial Fulfillment of the Requirements for the Degree of
Master of Engineering in Electrical Engineering and Computer Science
at the Massachusetts Institute of Technology

February 11, 1997

Author____
Department of Electrical Engineering and Computer Science
February 8, 1997

Certified by____
Charles M. Oman
Thesis Supervisor

Accepted by____
Arthur C. Smith
Chairman, Department Committee on Graduate Theses

Microgravity Induced Changes in Horizontal Vestibulo-Ocular Reflexes of SLS-1 & SLS-2 Astronauts

by

Matthew A. Neimark

Submitted to the Department of Electrical Engineering and Computer Science
February 11, 1997

In Partial Fulfillment of the Requirements for the Degree of
Master of Engineering in Electrical Engineering and Computer Science
at the Massachusetts Institute of Technology

Abstract

Data from two vestibular experiments conducted on the US Space Shuttle Space Life Sciences (1991, 1993) missions were combined to assess the effects of the microgravity (μG) environment on the horizontal angular vestibulo-ocular reflex (VOR) in a group of seven astronaut subjects. Data was collected during preflight, inflight, and postflight sessions. Eye movement was recorded via electrooculography (EOG) as the subjects were rotated at 120°/s for one minute, and then stopped. Recording continued for one minute after the chair stop. For one half of the runs, subjects pitched their head forward 90° immediately after the chair stop, which is known to "dump" post-rotatory nystagmus in 1-G. Slow phase velocity (SPV) was calculated via differentiation and order statistic (OS) filtering. First order model gains and time constants (τ) were calculated for each of the per-and post-rotatory run segments via a quadratic optimization method.

Since gain data was variable and showed no clear trends, analyses focused on time constants. Significant changes in inflight (FLIGHT) τ could not be demonstrated relative to preflight (PRE) for the entire group of subjects. Mean per-rotatory (PER) was 12.8 s; mean post-rotatory head-erect (HE) τ was 17.6 s. However, inflight data suggested a spectrum of inflight responses for individual subjects, and that those responses predicted the pattern of postflight responses:

Inflight testing indicated that the time constants of four of the subjects (Group II) were equal to or greater than preflight measures, and that the other subjects (Group I) had adapted head-upright τ's that were shorter relative to preflight values. Head-upright time constants clearly decreased (PER: by 3.0 s; HE: by 7.6 s) for Group I subjects between preflight and early postflight (EARLY) tests. Head-upright time constant decreases between preflight and late postflight (LATE) (PER: by 1.1 s; HE: by 2.5 s) occurred for the group of seven subjects.

Inflight dumping time constants increased relative to preflight and early postflight sessions (FLIGHT/PRE=1.3, FLIGHT/EARLY=1.5). In addition, significant decreases in inflight dumping τ's relative to head-erect inflight controls were not found, even in subjects with longer time constants, demonstrating that dumping is due to gravireceptive cue conflict.

This research was supported by NASA Contract NAGW-4333.

Thesis Supervisor: Charles M. Oman
Title: Senior Lecturer, Director Man Vehicle Laboratory

Acknowledgments:

I would like to thank my parents, Sheridan and Dana Neimark, not only for their most important roles in my up bringing, but for always being there, even though I have been away from home for some time. You've always been a pillar of support when I was in need. Thank you also, to my incredible sister, Rebecca Neimark, and brother-in-law Dr. Lee Spector, who have offered their home when I needed escape.

Thank you also, to my advisor, Dr. Charles Oman, who has given me, besides this wonderful project, an endowment of knowledge and scientific insight which could not be gained from any classroom activity. Thank you to Dr. Alan Natapoff, who spent hours helping guide me through statistical methodologies I used to support my results, and who also spent much time helping to proofread this Thesis. Thank you to Mr. Fred Amlee and the Life Sciences Data Archive team at JSC who have been safe guarding this precious data for future investigators.

Thank you to my predecessors Dave, Ted, and Chris. (We all know what Newton said.) Special thanks to Dave whose thesis was a real manual to the SLS-1 mission and data. Much thanks to Shelley who played a vital part in terms of helping to reanalyze the vast quantities of data.

Thank you to all the MVLers: Dave, Prashant, Patricia, Amir, Dawn, Jen, Adam, Christine, and Mike. Working along side you was a great experience, both social and professional. Best of luck in all of your futures.

Thank you very much to my academic advisor, Dr. Stephen Burns who has always offered much advice, encouragement, and support. Thank you also to other research advisors Drs. Michael Unser and Eugene Terray, who have given me research opportunities, and have helped me to become more skilled as an investigator. Many thanks to Professors Thomas Weiss and Gregory Wornell who have led the two most inspiring courses I took at MIT. Many thanks to Professor Alvin Drake who gave me an opportunity to improve my skills as an instructor.

Thank you also to my friends I made at MIT throughout the years: Ray, Manish, Jason C.: You helped ease my separation anxiety. Jason, Doris, Marcus, and Min: You've offered continued friendship and support. Matt, Petrina, Binky, Mark, and Sophie: You have offered interesting conversation and have helped shaped me into the music connoisseur I am. Andrea, Mike, Jose, and Martin: You have made MIT a fun place to go to grad school. You have also increased my international sophistication, and cultural sensitivity. Bram, Kendra, and Jackie: Although we haven't known each other for a long time, it's been nice living with you.

This research was supported by NASA Contract NAGW-4333 "Spacelab Rotating Chair Extended Data Analysis." Human subject research protocols had previously been reviewed by Institutional Review Boards at MIT and NASA-JSC.

Table of Contents

List of Figures

List of Tables:

Introduction

Background

One of the challenges associated with long duration space flight is the control of space motion sickness (SMS) that occurs in some astronauts after prolonged exposure to weightlessness. It is hypothesized that SMS arises from an unfamiliar combination of cues from the vestibular system and other (visual, tactile, proprioceptive) sensory systems. Studies of microgravity (μG)-induced changes in the vestibulo-ocular reflex (VOR) provide key insight into the state of the vestibular system, and into the origins of sensory conflict and SMS.

Non-voluntary eye movements, which help stabilize images on the retina during head rotation, are affected by both visual and vestibular cues. The central nervous system (CNS) processes the image, and this helps eye movement control through an attempt to minimize movement of the image on the retina..

If the head movement is rapid or unexpected, the visual processing becomes too slow and eye movement control becomes dependent upon the vestibular system. The VOR corresponds to eye movement arising totally through vestibular cues.

If a head rotation causes large eye movement, the eye will exhibit a nystagmus, an eye motion consisting of a slow, compensatory "slow phase" movement in the direction opposite of head rotation, followed by a quick movement, called a fast phase, in the direction corresponding to the head rotation. The slow-phase velocity (SPV) is the eye velocity signal during compensatory eye movement.

VOR responses can thus be experimentally elicited by rotating a subject while masking visual cues. Often in practice, this is done by placing a subject on a rotating chair in the dark. When the chair's motion stimulus is a rectangular pulse of rotational velocity, the subject's slow phase movement rises rapidly at first to a rotational velocity opposite in direction to that of the chair. The temporal evolution of the SPV during the pulse corresponds approximately to a decaying exponential. When the chair stops, the slow phase velocity will increase again, but in the opposite direction, and then decay again to zero. Measurements during chair movement are "per-rotatory" and those conducted during the period after chair has stopped are "post-rotatory."

The relationship between chair velocity input and SPV output is well approximated by a linear time invariant system. System models incorporate the effects of cupula endolymph dynamics and neural pathways which contribute to the VOR response. Such models have included up to three poles and four zeros (Raphan, Matsuo et al. 1979).

The simplest such model corresponds to a first order high pass filter: $H(s) = \frac{K\tau s}{\tau s + 1}$ (Steinhausen 1933; Van Egmond, Groen et al. 1949). In this case, K is the VOR gain and τ is the dominant time constant of nystagmus decay. Higher order models have been developed to describe neural adaptation, which is known to cause the observed undershoot in the SPV curves at the ends of the per- and post-rotatory periods (Figure 1). As a result of the undershoots, time constants estimated during the per-rotatory period of such a chair experiment run are invariably shorter (by ~5 seconds) than the time constants of the corresponding post-rotatory phases. Since these two time constants are predicted by the same model, their numerical values are highly correlated.

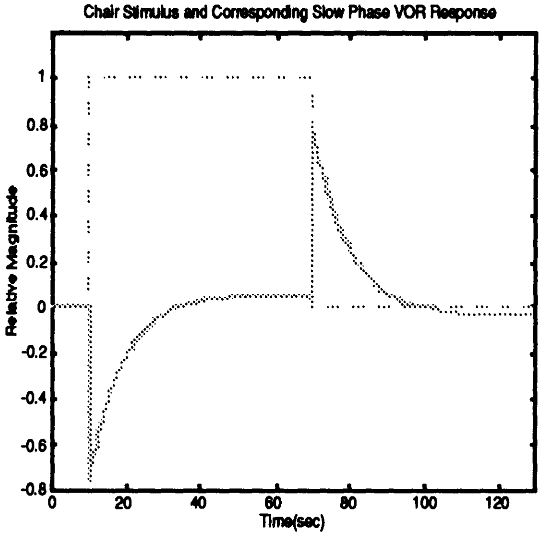


Figure 1. Idealized Rotating Chair Subject's SPV (solid) resulting from a one minute pulse of angular velocity (dashed) [from Balkwill, 1992]

Although the dynamics of the VOR decay were traditionally attributed to fluid flow in the semicircular canals of the inner ear. It was noted that people who were normally subjected to unusual vestibular stimuli (e.g. skaters, pilots) had VOR responses for which τ was considerably smaller. Studies on monkeys revealed that nystagmus persisted even after action potentials from the semicircular canals ceased. This suggested a neural mechanism with lead/lag dynamics that perpetuated the VOR.

There have been several higher order LTI system models that have accounted for this observed lengthening. One of the earliest (Robinson 1977) explains lengthening through visual system processing of a hypothetical efferent estimate of head rotation. The observed first order exponential nature of the VOR is explained through pole-zero cancellations.

Another model explains time constant lengthening through a hypothetical neural mechanism known as known as velocity storage (VS). VS is thought to act through an "indirect pathway" as a leaky integrator and has the effect of canceling the pole associated with the semicircular canal fluid flow, and substituting a longer time constant τ. This is accomplished, in terms of the system dynamics, through creation of a zero in the s-plane due to the indirect pathway and creation of a pole due to the leaky integrator. This zero cancels the pole associated with the canal time constant, yielding the VS pole as that associated with the VOR time constant (Raphan, Matsuo et al. 1979). Figure 2 shows a block diagram of the Raphan-Cohen model for VOR velocity storage.

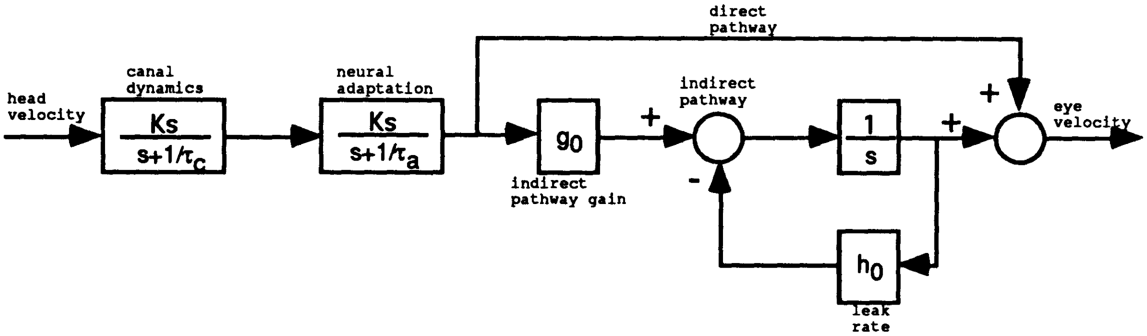


Figure 2. Block diagram of the Raphan-Cohen model for VOR velocity storage [from Balkwill, 1992]

The magnitude of the dominant time constant, τ, is affected by many factors including canal, otolith, visual, and proprioceptive cues. It is thought that the adaptive nature of the time constant serves to help process lower frequency angular motion dealt with in everyday life (Raphan, Matsuo et al. 1979).

Previous research has suggested that when subjects in 1-G perform a 90° pitch-forward head motion at the beginning of the post-rotatory period, the VOR time constant is shortened. This result, called "nystagmus dumping," seems to result from sensory conflict. Semicircular canal afferents indicate rotation about an axis perpendicular to the gravity vector, whereas otolith and tactile cues indicate that the head is immobile with respect to gravity. It is hypothesized that since VS attempts to integrate otolith, canal, and tactile cues, and since these cues provide conflicting information during the dumping experiment, VS is suppressed.

In 0-G, otolith cues which normally detect the presence of gravity are not present and cause sensory conflict due to their unfamiliar nature and partial disagreement with other cues (visual, semicircular canal, proprioceptive). The post-rotatory VOR time constant has been studied in 0-G parabolic flight, and was found to decrease significantly, both for subjects as a group (DiZio and Lackner 1988) and also in most individual subjects (Oman, Pouliot et al. 1996). Forward (Oman, Pouliot et al. 1996) or backward (DiZio and Lackner 1988) head movements did not further shorten the time constant by a significant amount. It was suggested (DiZio and Lackner 1988; Oman, Pouliot et al. 1996) that the shortening of the time constant arose from loss of VS due to the unfamiliarity of otolith cues after transition to 0-G. Since velocity storage was lost with the head erect, it could not be determined whether head tilt also triggered dumping (Oman, Pouliot et al. 1996).

Astronaut subjects exposed to the μG of spaceflight should likewise demonstrate this suppression of VS. A question, however, arises as to what happens to the VOR VS after inflight adaptation occurs. Some subjects might show VOR with increased τ since their VS has adapted to their low frequency motions made in μG. Still other subjects might adapt with lower τ because their VS might be ignoring vestibular cues and utilizing more visual and tactile cues (to minimize sensory conflict). An earlier analysis of SLS-2 rotating chair data suggested that individuals may adapt differently to prolonged μG, so that some subjects show a persistent loss of VS, while others show normal or even prolonged VOR time constants (Oman, Pouliot et al. 1996).

It has been hypothesized that (Oman and Kulbaski 1988; Oman and Balkwill 1993; Oman, Pouliot et al. 1996) for a brief time after return to earth, the VS mechanism should also be inactivated until readaptation to the original environment takes place. In both cases, while velocity storage is inactivated, time constants should be shortened since the VOR time constant reverts to that of the canal dynamics. Previous studies have shown that after other types of sensory disturbances, VOR readaptation normally takes place from one to two weeks (Gonshor and Jones 1971)

Experiments performed over the past decade have clarified μG's effect on the VOR but have been inconclusive regarding certain phenomena. Questions regarding inflight dumping, VS adaptation inflight, and readaptation after return 1-G remain to be answered. This study attempts to demonstrate conclusively μG's effect through the integration of data from two rotating chair experiments performed on two Space Shuttle missions: Space Lab Sciences (SLS)- 1 (1991) and SLS-2 (1993). The experimental methodologies corresponding to the two missions were practically identical in that subjects' VOR were recorded in a similar manner before, during, and after the spaceflight. Pooling data from these two missions increases the number of subjects in the study, and therefore its statistical power.

Previous Results

The effect of prolonged weightlessness on the postflight VOR has been studied in five previous space shuttle spaceflight missions (SL-1, D-1, SLS-1, IML-1, SLS-2). Three of these missions (SLS-1, IML-1, and SLS-2) also yielded usable inflight data. All of these missions employed similar protocol for eliciting responses (one minute, 120°/s, rectangular rotating chair stimuli) and measured horizontal VOR responses via EOG during the per- and post-rotatory periods. Much has been learned concerning VOR changes as a result of exposure to μG over the course of these missions. In addition, processing techniques for rotating chair data have increased in sophistication since the first SL-1 analysis and have made it possible to detect changes in VOR responses with greater accuracy and greater efficiency. Both scientific findings observed in the previous mission analyses and the analytical techniques used to determine these findings will be briefly reviewed:

Scientific Results: The SL-1 mission (1983) experiment incorporated four subjects tested over five preflight and three postflight sessions. The first analysis of this data (1988) computed SPV using a computerized method and estimated τ and K through log-linear regression. It found that three of the four subjects had decreased τ (by 21%) during the first two postflight sessions (one and two days after landing), but concluded that by the fourth day after landing, τ had returned to the preflight value (Oman and Kulbaski 1988).

A second analysis (1993) of the SL-1 mission again estimated the first order model parameters. These estimations, however, were performed through constrained optimization (Grace 1990; Liefield 1993). Order statistic (OS) filtering (Engelken and Stevens 1990; Liefield 1993) was used instead of an acceleration detection program to estimate the SPV decay envelope. In addition, data corresponding to the subject whose time constant did not significantly change postflight in the previous SL-1 analysis were not included in this analysis due to small EOG potentials. Not surprisingly, the results of this study determined that all three subjects had decreased time constants postflight. However, by this analysis, none of the subjects demonstrated that τ had recovered preflight values by the fourth day after landing (Liefield 1993).

The subsequent D-1 mission (1985) also included a pre- /postflight rotating chair experiment. However, model-fitting techniques were not used to analyze its data. Instead, the SPV time series were averaged by subject, direction, and session. It turned out that two of the five subjects had direction asymmetries. Of the three who did not, all preflight SPV responses were averaged and compared to the average of all the responses from the first postflight session (first day after landing), and through a χ^2 test, it was determined that postflight responses decayed more rapidly (Oman and Weigl 1989), from which one might infer that the VOR time constant shortened.

The rotating chair protocol of the SLS-1 (1991) mission was similar to the previous two, but also measured responses inflight (on days four and five after launch) and included for the first time a dumping experiment during which the head was pitched forward after chair stop and held in this position through the entire post-rotatory period. The first analysis of the SLS-1 data indicated that mean inflight post-rotatory head-erect time constants, estimated through log-linear regression, were similar for 3/4 subjects compared to preflight. Dumping inflight time constants also increased for the same three subjects, presumably because the otolith organs which cause sensory conflict were not stimulated inflight. An early postflight decrease in head-erect time constants was detected only for one subject (Oman and Balkwill 1993).

The rotating chair experiment performed as part of the IML-1 (1992) mission also included inflight measures, but dumping head movements were not possible due to equipment limitations. The analysis determined that subjects' time constants decreased inflight, and returned to normal postflight (Oman and Calkins 1993). One explanation for the fact that postflight decreases in τ

were not detected is that time constants were averaged across both early and late postflight sessions (which occurred on the mission return day, and on the first, second, fifth, eighth and 30th day thereafter).

The SLS-2 (1993) experiment was similar to the previous four. However, the analysis employed quadratic optimization as a means to produce improved estimates of K and τ as opposed to log-linear regression. Data was collected pre-, in-, and postflight. The inflight measures were conducted both early (4 days) and late (10 days) into the mission. The analysis distinguished early (first three experimental sessions) from late (latter three experimental sessions) postflight. Data on the acute effects of 0-G were obtained in parabolic flight for 3 of the 4 subjects. The study found that after several days in orbit, τ with head-upright for the inexperienced astronaut subjects had a tendency to increase inflight whereas those for the experienced subjects remained lowered (compared to preflight). In addition, for both subjects tested on the second flight session, head-upright τ tended to increase during the mission, suggesting VS restoration. The study also found a significant decrease in head-upright τ during early postflight as compared to preflight.

Although changes in K across gravity conditions have been investigated in the previous experiment, these changes have been inconsistent and highly variable. However, the IML-1 and SL-1 experiments suggested that gains were reduced postflight compared to preflight, and the IML-1 experiment also detected an inflight increase in K.

Previous Data Analysis Techniques: For the first analysis of SL-1's data, fast phase removal was accomplished by an eye acceleration detection algorithm (Massoumnia 1983). This algorithm would detect the beginning and end of a fast phase, and attempt to interpolate SPV through it. Unfortunately, the algorithm was unable to detect a considerable number of fast phases necessitating manual editing. This process was time consuming and it introduced human error and potential bias into the experimental results.

For the D-1 analysis, the raw eye-position data was analog, and was electronically differentiated. Sampling and SPV envelope detection was performed manually from chart records, and through manual sampling of electronically differentiated analog eye movement signal (Oman and Weigl 1989).

Order statistic (OS) filtering was introduced in the SLS-1 analysis as a means of automatically interpolating through fast-phases. OS filters are running-window non-linear filters whose outputs are based upon ordered statistics (such as the median) of data in the window. Two such OS filters were utilized: The predictive finite impulse response mean hybrid (PFMH) filter, and the adaptive asymmetrically trimmed mean (AATM) filter. The PFMH filter operated on the raw eye position signal and had the effect of smoothing noise artifacts and sharpening nystagmus peaks to facilitate detection. The AATM filter operated on the eye velocity signal, and interpolated through the fast phases by estimating a centroid in a truncated histogram of the window (Engelken and Stevens 1990).

Characterization of the VOR's response decay has been accomplished via two means in the previous experiments: One method, denoted parametric analysis, involves identification of K and τ through model fitting, averaging within various experimental conditions, and subsequently determining effects of the various conditions via statistical comparisons of the averaged parameters. Parametric analysis was utilized in both SL-1 analyses, the IML-1 analysis, and the SLS-2 analysis.

Another means of determining effects of the various conditions, denoted ensemble averaging, is to average entire run segment time series within the various conditions. Differences between the time series averages under the various conditions is then determined through statistics such as χ^2 and Σt^2 (Oman and Kulbaski 1988; Oman and Weigl 1989; Liefield 1993; Oman and Balkwill 1993; Pouliot 1995).

Parameter identification itself has been realized by different methods: One such method, called log-linear regression, involves operation of the logarithm function on a run segment (either per- or post-rotatory portion of the run) and then performing linear regression on this transformed time series. Log-linear regression is relatively simple and quick to perform, but it has a tendency give greater weight to lower magnitude data points in the SPV waveform, which also have lower signal to noise ratios.

The other method, called constrained optimization, involves determination of the global minimum of a multiparameter function which is the mean square difference between the data time series and the model time series. For models larger than first order, this method is more appropriate (Grace 1990; Balkwill 1992; Liefield 1993). MATLAB function and scripting abilities have made it possible to analyze entire sessions of a subject's data in a semi-automated manner reducing the processing time (Balkwill 1992; Liefield 1993; Pouliot 1995). Constrained optimization can be used to fit either data from individual runs, or ensemble averaged data.

Goals of This Study

The evolution of rotating chair studies performed during the past two decades have contributed to our knowledge of vestibular adaptation in space. However, analyses of data from individual missions have been limited in that the number of subjects participating in each of the studies have been small. Furthermore, analytical methods have improved over the course of the missions.

It is thus advantageous to combine data from these studies such that a large group of subjects can be studied simultaneously and such that the most sophisticated and pertinent analytical techniques can be used consistently for all of the subjects.

Ultimately, one would like to combine all the data from the five missions for a grand study of μG's effect on the VOR. Inflight data was available from only three missions. The procedures used for testing on SLS-1 and SLS-2 were virtually identical, so this study concentrated on those. It was hoped that by combining data from the two missions, one could search for trends in the data with a fresh perspective, and have the ability to draw better conclusions as to how subjects, in general, react to the gravity changes.

One of the goals of this study was to determine whether or not SLS-2 conclusions regarding head-erect inflight data were confirmed when the subject population was enlarged from four subjects to eight. The previous SLS-2 analysis grouped subjects as "slow adapters" and "fast adapters" based on inflight head-erect measures of VOR time constants. The SLS-2 study noted that there was an apparent correlation between this subject grouping and previous flight experience of the subject which also appeared to apply to the earlier missions (Oman, Pouliot et al. 1996).

Another goal was to verify the previous finding of decreases in head-upright τ between inflight and early postflight testing sessions. Different rotating chair studies have provided different conclusions regarding this comparison: The SL-1 (both analyses), D-1, and SLS-2 studies suggest the existence of this decrease, whereas the SLS-2 and IML-1 studies did not find a decrease for all subjects.

A further goal of this study was to determine whether or not head-upright τ regained baseline values by late postflight testing. Although the first analysis of the SL-1 data indicated that this was the case, the second analysis was unable to confirm recovery. The SLS-1 and SLS-2 missions could not find a difference between preflight and late postflight time constant values. The IML-1 study did not distinguish between early and late postflight testing.

A fourth goal of this research was to confirm the observation in both SLS-1 and SLS-2 that time constants were not reduced by the dumping maneuver during inflight testing, even in those subjects who had apparently recovered VS.

Methods

Experimental sessions were conducted before, during, and after the SLS-1 (STS-40; June 5-14, 1991) and SLS-2 (STS-58; October 18-November 1, 1993) spaceflight missions. Subjects in this study included five males and three females. Three of the subjects were Payload Specialists, and five were NASA Mission Specialists. Four of the subjects were making their first spaceflight while the other four had flown before. The subjects are here coded G, M, N, P, T, V, X, and Y (M, N, P, T correspond to SLS-1 and G, V, X, Y correspond to SLS-2). Two of the subject codes correspond to an individual who flew on both spaceflights. While none of the subjects had history of vestibular disease, one subject had esophoria, reduced acuity in the right eye, and childhood strabismus surgery in that eye.

The experimental design of this combined study is factorial with repeated measures, so that each subject served as his or her own control. For both missions, four 1-G preflight (PRE) sessions were conducted at the NASA Johnson Space Center (approximately 130, 90, 60, and 25 days before launch for SLS-1, and 122, 110, 88, and 18 days before launch for SLS--2). For SLS-1, inflight (FLIGHT) testing consisted of one session conducted on the fourth flight day for subject N and the fifth day for the three other SLS-1 subjects. The inflight testing for SLS-2 consisted of two sessions conducted on flight days 4 and 10. Postflight testing on SLS-1 was conducted over four sessions (one, two, four, and seven days after landing for subject N, and on the landing day and one, four, and seven days thereafter for the other subjects). The first two sessions have been designated as "early" postflight (EARLY) and the latter two as "late" postflight (LATE). Postflight testing for the SLS-2 experiment was conducted over six sessions (on the landing day and one, two, six, nine, and 11 days thereafter). Similar to the SLS-1 experiment, the first three postflight sessions have been designated "early" and the latter three have been designated "late".

The rotating chair employed in the ground experiments was motorized and velocity-servo controlled (Oman and Balkwill 1993). This chair rotated on an axis parallel to the gravity vector.

Subjects placed in the chair wore light occluding eye goggles to mask visual cues, and wore binaural earphone monitors through which the operator communicated in order to mask auditory rotation cues and maintain alertness.

The chair employed during orbital experiments was lightweight, and manually spun by an operator standing beside it in foot loops. Desired constant rotational speed was achieved by pushing the chair three times per rotation in a smooth 1-Hz metronome cadence. Subjects were seated in this chair with their legs crossed in a lotus position. The subjects wore goggles identical to those in ground testing. On SLS-2 they wore binaural headphones that actively canceled noise, but on SLS-1, wore earplugs.

Each experimental session consisted of a series of rotation tests, denoted as runs. Ideally, there were 8 runs per ground or SLS-2 inflight session. Normal SLS-1 inflight sessions consisted of 5 runs. Some factors limiting sessions from incorporating the ideal number of runs included computer crashes, subject nausea, and lack of time necessary for a session's completion. The normal testing sequence is listed in Table 1. SLS-1 inflight testing incorporated slightly different sequences.

Table 1. SLS-1 ground & SLS-2 ground, inflight experimental test sequence

EOG CAL
CW HE
CCW HE
CW DMP
CCW DMP
EOG CAL
CW HE
CCW HE
CW DMP
CCW DMP
EOG CAL

EOG, electrooculogram; CAL, calibration run; CW, clockwise run; HE, postrotatory head-erect run; CCW, counterclockwise run; DMP, dumping run

A run consisted of two phases: The chair, initially at rest, accelerated to 120°/s in a manner approximating a step in either a clockwise (CW) or counter-clockwise (CCW) direction. After one minute of rotation at constant speed, the chair was similarly decelerated to a stop, while measurements were still performed at least up to one minute after this event. (The electrically powered chair's acceleration was controlled by a shaping circuit and full acceleration and deceleration were achieved within 0.5 s. The manually powered chair's acceleration and deceleration were only slightly lower).

There were two different types of runs conducted in the sessions: head-erect and dumping. During the head-erect runs, the subject's head was upright and centered over the chair rotation axis during both the per- and post- rotatory phases of chair motion. For a dumping run, the subject pitched his or her head forward (mostly at the head and shoulders) forward 90° just after the chair stopped.

Electrooculography (EOG) was utilized to measure binocular horizontal eye position in both ground and inflight testing. The EOG bi-temporal neonatal surface electrodes were pregelled and placed on the subjects' outer canthus. The measured signals were amplified (various amplifications were employed dependent on mission and testing location) (Balkwill 1992; Pouliot 1995). The amplified signals were then lowpass filtered (ground: 30 Hz cutoff; inflight: 40 Hz cutoff) and subsequently digitized and sampled (ground: 120 Hz sampling frequency; inflight: 100 Hz sampling frequency). EOG calibrations were performed via separated wall targets (ground separation: 23.4°; inflight separation: 18°) at which the subjects gazed.

The EOG signals were segmented by run. The MATLAB numerical software package (version 4.0, Natick MA) was used to analyze these time series.

Session calibration factors were calculated from corresponding calibration runs through semiautomatic software (Balkwill 1992). After scaling with the appropriate calibration factor, these signals were then filtered twice by a PFMH OS filter in order to smooth noise artifacts and sharpen nystagmus corners. This output was subsequently passed through a differentiation/low pass (convolution of a 3 point differentiation and 9 point Parks-McClellan 10 Hz corner frequency low pass filter) to determine eye velocity. The resultant nystagmus signal, which consisted of alternating fast and slow phases, was then passed through an AATM OS filter (1 s window, α=0.44, β=0.12, γ=0.4) to interpolate through the fast phases, yielding the SPV.

The data from the runs was further segmented into the per- and post- rotatory portions from tachometer records which indicated when the chair started and stopped. Segments are defined as time series data corresponding to either of these portions. Segments were either zero padded or truncated to 60 s. Per-rotatory segments will henceforth be abbreviated "PER," post-rotatory head-erect segments will be abbreviated as "HE," and post-rotatory dumping segments will be abbreviated as "DMP." "Head-upright," or "upright" segments are defined as those corresponding to PER or HE.

Dropouts and wild points were removed from the time series via a recursive log-linear regression technique: Each data segment was logarithmically transformed, and the period of the signal between 1 s after chair start/stop to the point at which the average SPV is less than 10°/s (within a five second window and at least 20 s after the chair start/stop) was fit with a straight line. Residuals greater than 6 times the RMS error were designated as wild points and removed from the time series. Dropouts were similarly designated as residuals greater than three times the RMS error and below 7.4°/s (2 ln SPV units) which were also removed. This regression/outlier detection procedure was repeated on the remaining data until the RMS error converged to within 20% the previous value (Balkwill 1992).

The run segments, with outliers removed, were subsequently resampled (using MATLAB's "resample" function) to 4 Hz. The period of the segments from 3 s after chair start/stop were then fit with the exponential VOR model using a constrained optimization procedure (Grace 1990).

Certain run segments were excluded from the final analysis because there was either a suppression of the VOR (from fatigue, for example) or the amount of data removed from time series during outlier detection rendered parameter identification inaccurate. The following criteria were employed in exclusion of run segments from the final analysis: 1) A constraint in the model was reached ($K \leq 0.06$, $\tau \leq 0.15$), 2) greater than 40% of first 25 s of the SPV data had been removed during outlier detection, 3) The MSE regression error divided by the model gain (MSE/K) exceeded 500(°/s). These exclusion criteria have remained unchanged from the previous SLS-2 analysis except for the third. For the original SLS-2 analysis, a segment was excluded if its corresponding MSE exceeded $200(°/s)^2$. This criterion was changed to accommodate the fact that SPV signals with higher average amplitude often have higher average MSE.

Parameter identification methods were similar to those performed for the SLS-2 analysis (Pouliot 1995). Differences were: 1) Usage of MATLAB 4.2 as opposed to MATLAB 3.0, 2) Recalculation of session calibration factors, 3) Differentiation filter implemented through MATLAB code as opposed to through the MEX C interface program previously used, 4) Ideal low pass filtering signal prior to resampling as opposed to averaging decimated values (Pouliot 1995).

Because time series measures in an appropriate format were unavailable for SLS-2 inflight data, parameters estimated during the previous SLS-2 analysis were used instead. (Almost perfect correlations were found between τ's estimated in this study and corresponding time constants estimated in the previous SLS-2 analysis.)

Analyses of variance (ANOVAs), repeated measures ANOVAs, t-tests, post-hoc tests of effects, and various non-parametric analyses (sign test, Friedman test) were performed with a statistical package (SYSTAT 5.2.1 for Macintosh). Manual calculations (using MATLAB 4.2) were performed for some of the non-parametric tests (one sided sign test, multiple comparison Friedman analysis).

Results

The dependent variable for the following analyses is the estimated long VOR time constant τ. Independent variables include subject, segment type (per-rotatory, post-rotatory head-erect, or dumping maneuver=PER, HE, or DMP) and condition (preflight, inflight, early postflight, or late postflight=PRE, FLIGHT, EARLY, or LATE). Per-rotatory and post-rotatory segments considered together are denoted "head-upright."

The exclusions outlined in the previous section prevented the analysis of some segments. Table 2 shows the number of remaining segments, after exclusions, tabulated by subject, segment, and condition (Table 2).

Subject fatigue and lowered motivation have been implicated in the high proportion of discarded run segments in the SLS-1 and SLS-2 analyses (Balkwill 1992; Oman, Pouliot et al. 1996). Unfortunately, some data cells (e.g. subject M, per-rotatory early postflight=PER-EARLY) had no measures. This narrowed the scope of certain planned statistical analyses (e.g. simple repeated measures ANOVAs).

Table 2. Number of run segments included in analyses / measures attempted

SUBJ	PER PRE	PER FLIGHT	PER EARLY	PER LATE	HE PRE	HE FLIGHT	HE EARLY	HE LATE	DMP PRE	DMP FLIGHT	DMP EARLY	DMP LATE
G	1/16	6/16	5/16	13/24	2/8	3/8	4/8	3/12	8/8	4/8	8/8	7/12
N	32/32	5/9	13/13	15/15	14/15	1/3	8/8	8/8	14/15	4/6	5/5	7/7
T	16/31	0/6	11/16	5/16	10/15	1/2	2/8	2/8	8/15	1/4	4/8	7/8
X	10/32	12/16	3/16	0/0	4/16	7/8	1/8	0/0	5/16	8/8	1/8	0/0
M	17/32	2/5	0/15	6/16	8/16	1/2	1/15	4/16	0/16	1/3	0/0	0/0
P	16/29	1/5	7/12	5/16	7/16	0/2	0/6	2/8	0/14	3/3	0/4	0/8
V	28/32	11/16	12/16	13/24	12/16	5/8	8/8	8/12	8/16	6/8	6/8	8/12
Y	22/32	7/16	11/18	14/24	15/16	4/8	7/10	4/12	12/16	3/8	6/8	8/12

PER, per-rotatory; PRE, preflight; FLIGHT, inflight; EARLY, early postflight; LATE, late postflight; HE, postrotatory head-erect; DMP, postrotatory dumping maneuver

Tables 3 and 4 show, respectively, the means and standard deviations within each cell (i.e. each subject-segment-condition combination). Figure 3 shows plots of these values by subject.

Subjects G, T, and X show higher variability within cells than did the other subjects (Figure 3, right), as was noted in the original SLS-2 analysis (Oman, Pouliot et al. 1996).

Table 3. Time constant sample means

SUBJ	PER PRE	PER FLIGHT	PER EARLY	PER LATE	HE PRE	HE FLIGHT	HE EARLY	HE LATE	DMP PRE	DMP FLIGHT	DMP EARLY	DMP LATE
G	16.74	12.36	12.92	15.56	30.75	16.62	23.23	27.31	13.96	12.14	13.12	15.05
N	15.60	12.42	13.71	14.94	19.03	18.85	18.04	17.43	11.53	19.08	9.588	10.33
T	20.29	-	16.60	18.20	28.15	17.71	26.27	25.27	15.46	23.00	18.85	13.83
X	11.30	7.531	8.817	-	20.17	14.63	8.050	-	15.14	11.01	7.790	-
M	12.46	12.94	-	11.15	15.49	19.87	13.27	13.40	-	11.77	-	-
P	11.68	18.46	11.33	10.60	15.96	-	-	12.93	-	18.10	-	-
V	9.350	11.99	9.391	9.751	13.58	13.44	11.69	11.47	7.611	13.56	6.300	7.248
Y	12.89	14.16	14.17	10.92	17.70	22.18	22.25	15.62	10.68	19.52	11.41	9.869

PER, per-rotatory; PRE, preflight; FLIGHT, inflight; EARLY, early; postflight LATE, late postflight; HE, postrotatory head-erect; DMP, postrotatory dumping maneuver; -, no data within cell

Table 4. Time constant sample standard deviations

SUBJ	PER PRE	PER FLIGHT	PER EARLY	PER LATE	HE PRE	HE FLIGHT	HE EARLY	HE LATE	DMP PRE	DMP FLIGHT	DMP EARLY	DMP LATE
G	#	9.854	2.262	2.796	0.827	10.48	4.249	9.335	1.429	3.013	3.428	1.635
N	1.743	3.111	1.377	1.770	1.718	#	0.999	3.080	2.085	2.451	1.848	2.247
T	4.735	-	3.729	5.610	5.801	#	1.853	4.533	2.378	#	6.081	3.378
X	2.219	3.326	0.967	-	6.311	6.891	#	-	6.502	4.979	#	-
M	1.706	1.358	-	1.351	1.463	#	#	3.119	-	#	-	-
P	2.112	#	1.642	1.718	1.727	-	-	0.148	-	3.718	-	-
V	1.425	2.481	1.414	0.840	2.313	2.423	1.565	1.824	1.620	3.126	1.355	1.358
Y	2.638	2.224	3.258	2.627	2.542	3.040	3.772	5.813	1.936	2.714	1.203	1.535

PER, per-rotatory; PRE, preflight; FLIGHT, inflight; EARLY, early postflight; LATE, late postflight; HE, postrotatory head-erect; DMP, postrotatory dumping maneuver; -, no data within cell; #, only one data point within cell

Subjects' mean gains ($\overline{K}$) were also calculated by segment. $\overline{K}$ =(0.70, 0.64, 0.73, 0.75) for per-rotatory (PRE, FLIGHT, EARLY, and LATE). $\overline{K}$ = (0.61, 0.58, 0.67, 0.60) for post-rotatory (PRE, FLIGHT, EARLY, and LATE). Pooled standard deviations for K were (0.18, 0.25, 0.16, 0.18; 0.16, 0.12, 0.12, 0.15) for (PER; HE) segments during (PRE, FLIGHT, EARLY, and LATE) sessions. This gain data was highly variable, and suggested no clear trends across conditions.

In the following sections, orbital and postflight results are presented in the following order: 1) Inflight Head-Upright, 2) Early Postflight Head-Upright, 3) Late Postflight Head-Upright, 4) Dumping.

Results will usually be presented first for the entire set of eight subjects, then for groups within the population of subjects, and finally for individual subjects.

Results will usually be presented first for the entire set of eight subjects, then for groups within the population of subjects, and finally for individual subjects.

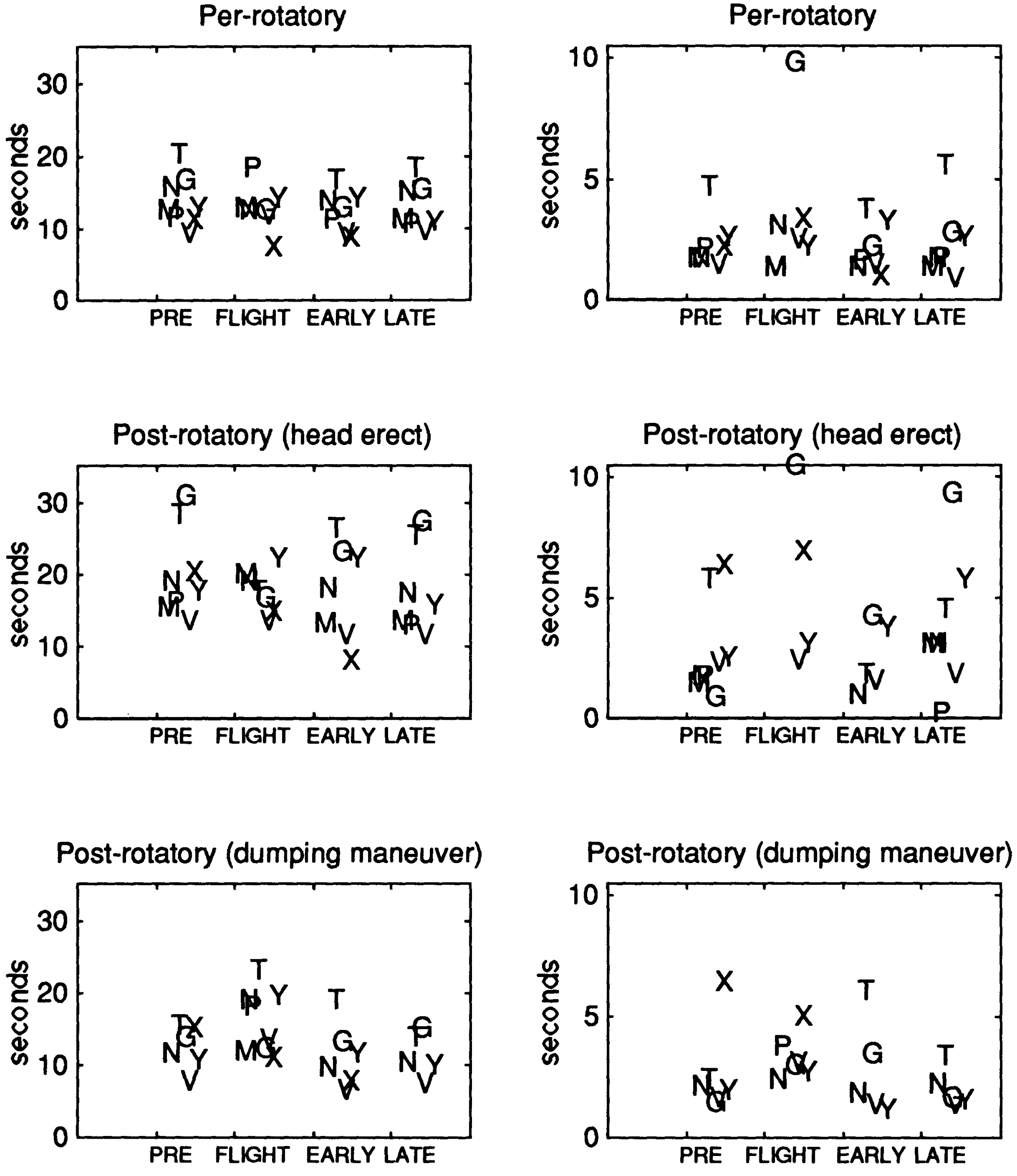


Figure 3. Cell means (left) and standard deviations (right) for τ within subjects, segment types, and experimental conditions

Inflight Head-Upright Results:

Figure 4 suggests no consistent trend across all subjects either to increase or decrease their head-upright (PER and HE) time constants from preflight to inflight sessions. Instead, as the original SLS-2 study found, some subjects' time constants decreased and others stayed the same or increased. Thus, as in the original SLS-2 analysis, subjects were placed in two categories: For Group (I; II) containing subjects (G, N, T, X; M, P, V, Y), respectively, the mean head upright time-constants inflight were (less; equal to or greater) than in preflight control tests. The SLS-2 subjects divided in the present analysis as they did in the original SLS-2 study. (The subject who flew on both missions fell into the same category under both analyses.) DMP results will be discussed later.

In particular, Group I's mean inflight τ was (significantly) 3.8 s (26%) shorter (paired t=10.976; df=2; p=0.008) and 7.6 s (31%) shorter (paired t=2.502; df=3; p=0.088) than corresponding preflight measures for per-rotatory, and post-rotatory head-erect segments respectively. These decreases had been previously observed (Oman, Pouliot et al. 1996).

Mean per-rotatory and post-rotatory head-erect time constants of Group II subjects in general increased during inflight testing (compared to preflight controls). Unlike Group I's results, Group II's paired t-tests did not suggest a statistically significant difference.

Table 5 shows independent-sample t-test (separate-variance) results for the differences between preflight and inflight sessions (by subject and segment type). T-testing indicated that half of the differences were significant for the PER segment comparisons, whereas none were significant for HE. For example, these values indicate a per-rotatory decrease of Group I subject X between preflight and inflight sessions (p=0.005) and a per-rotatory τ increase of Group II subject V (p=0.006). Group II subject Y had a post-rotatory head-erect time constant increase that was also significant ($p=0.051$). (Only subject X's decrease was significant when Bonferroni-corrected for post-hoc effects, $p<0.05/9=0.0056$.)

The previous SLS-2 study, pooling results from the SLS-1, SLS-2, and IML-1 studies ($n=11$), suggested that there was a correlation between a subject's category (I; II) and his/her previous flight experience (Oman, Pouliot et al. 1996). In the present study, three of the four experienced subjects had decreased inflight τ's while three of the four inexperienced subjects had increased or similar inflight τ's. (One subject flew in both missions.) In all, the category (I; II) of five of the seven subjects can be predicted by previous flight experience. This, however, is not significant (one sided sign test; $p=0.23$), but (because previous experience is expected, a priori, to play a major role) it is a highly suggestive result.

As opposed to the post-hoc categories, Groups I and II, subjects were placed in two a priori categories based on experience: Subjects within (Group 1; Group 2) (had flown before; were inexperienced). Although (Group 1; Group 2) means tended to (decrease; increase) in the expected direction, paired t-tests, of PER and HE τ's within Groups did not show statistically significant differences.

Early Postflight Head-Upright Results:

Head-upright time constants were reduced immediately after return to earth gravity. Fifteen of the twenty comparisons (for individual subjects and conditions) showed mean decreases between preflight and early postflight. By segment, (5 of 7, 6 of 7, and 4 of 6) within (PER, HE, and DMP) showed decreases. That is, 11 out of 14 of the head-upright means decrease between preflight and early postflight (one sided sign test, $p=0.0286$). (The one sided test was used since it had been expected that early head-upright postflight measures would decrease from the preflight controls) (Oman and Kulbaski 1988). Neither the sign test, nor the other non-parametric statistical tests, assume normality or equal variance of the subject populations tested.

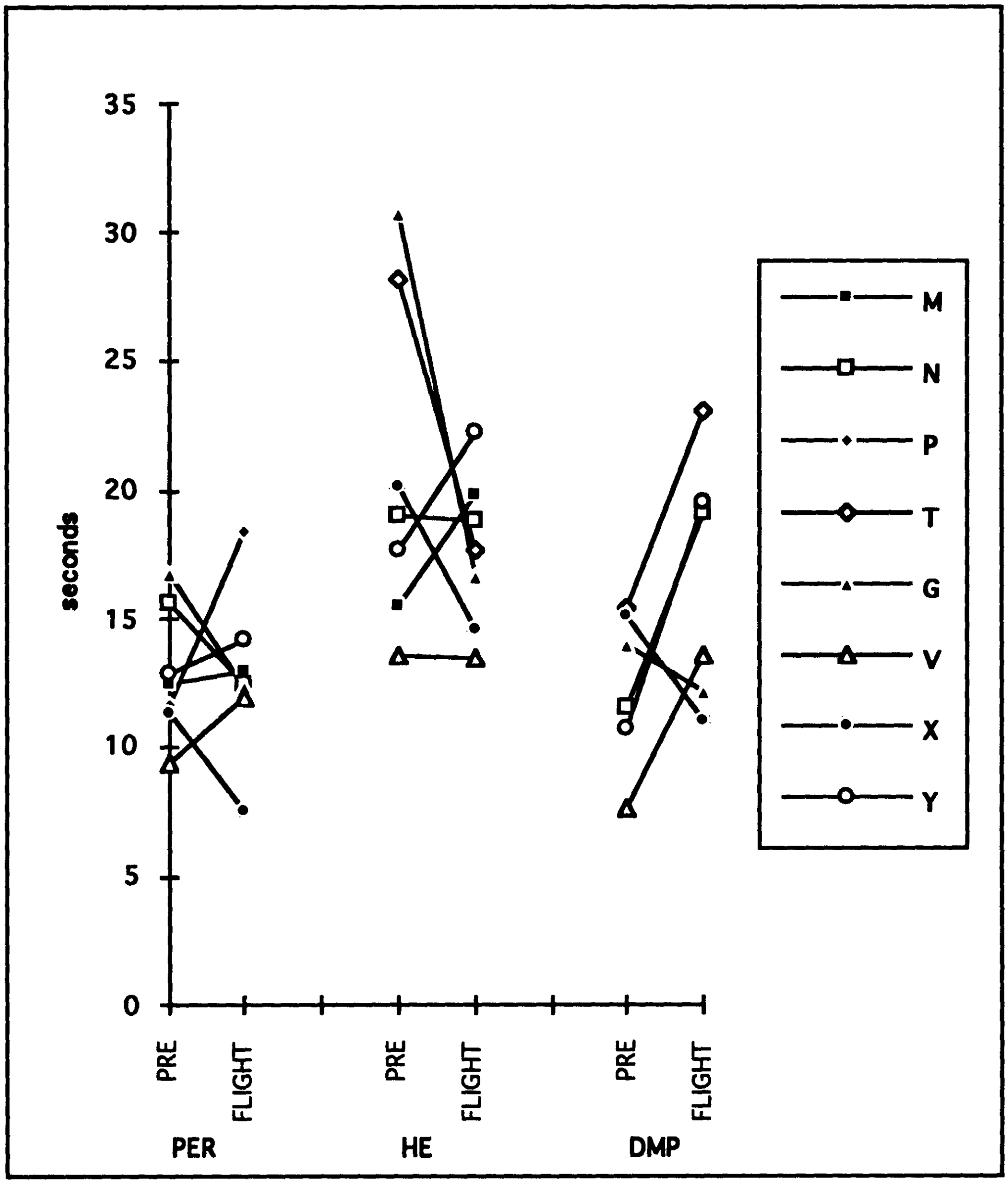


Figure 4. Changes in mean time constant for per- and post-rotatory (head-erect and dumping) segments between pre- and inflight testing sessions

Table 5. t statistics for τ comparisons between pre- and inflight testing

	PER			HE			DMP		
SUBJ	t	dof	p≤	t	dof	p≤	t	dof	p≤
G	-	-	-	2.325	2.0	0.143	1.145	3.7	0.321
N	2.232	4.4	0.083	-	-	-	-5.612	4.3	0.004
T	-	-	-	-	-	-	-	-	-
X	3.176	19.2	0.005	1.355	6.9	0.218	1.213	6.9	0.218
M	-0.459	1.4	0.707	-	-	-	-	-	-
P	-	-	-	-	-	-	-	-	-
V	-3.322	12.7	0.006	0.109	7.2	0.916	-4.258	7.0	0.004
Y	-1.253	11.9	0.234	-2.707	4.2	0.051	-5.312	2.5	0.020

PER, per-rotatory; HE, post-rotatory head-erect; DMP, post-rotatory dumping maneuver, - not enough data for a t-test; results for which $p \leq 0.05$ are underlined

The previous SLS-2 study suggested that adaptive mechanisms that contributed to inflight reductions in Group I would also be likely to cause reductions between preflight and early postflight sessions (Oman, Pouliot et al. 1996). Group I's per-rotatory values decreased 3.0 s=19% (on average) from preflight to early postflight (paired t=6.337; df=3; p=0.008). The same subjects' post-rotatory head-erect mean values showed no significant differences.

Corresponding comparisons for Group II showed no significant changes in either per-rotatory or post-rotatory head-erect time constants, suggesting that their time constants did not decrease as much as Group I's. In addition, Subject Y had mean τ increases from preflight to early postflight testing for all segment types. For post-rotatory head-erect segments, this increase was significant by an independent sample (separate variances) t-test (p=0.018).

Independent-sample (separate-variance) t-tests (Table 6) indicated that per-rotatory time constants of Group I subjects N, T, and X decreased between preflight and early postflight sessions (p=0.001, p=0.033, p=0.023 respectively). A similar test noted a significant post-rotatory τ decrease of Group I subject G (p=0.034). In Group II, only subject V showed a significant decrease in τ (p=0.042). (Subject N's decrease was significant even when Bonferroni-corrected for post-hoc effects, $p<0.05/11=0.00454$.)

Late Postflight Head-Upright Results:

Figure 6 compares preflight and late postflight sessions for each subject and segment. It has been thought that the values of τ should, by postflight, return approximately to their preflight control levels. In fact, the great majority (13 out of 14) of the subjects' head-upright postflight mean time constants are less than their corresponding preflight means (two sided sign test p=0.0017). This suggests that late postflight τ's are less (if only slightly so) than those measured during the preflight control sessions.

Paired t-tests found statistically significant decreases from preflight to late postflight in per-rotatory (PER) constant (1.1 s; 8%), (t=3.6, df=6, p=0.012), and in head-erect constant (2.5 s; 12%), (t=9.9, df=6, p<0.001). Some individual subjects of Group II showed head-upright increases by independent-sample, separate variance t-tests (Table 7): Subject P's post-rotatory head-erect decrease was significant (p=0.003) at the Bonferroni-corrected level (0.05/13=0.00385).

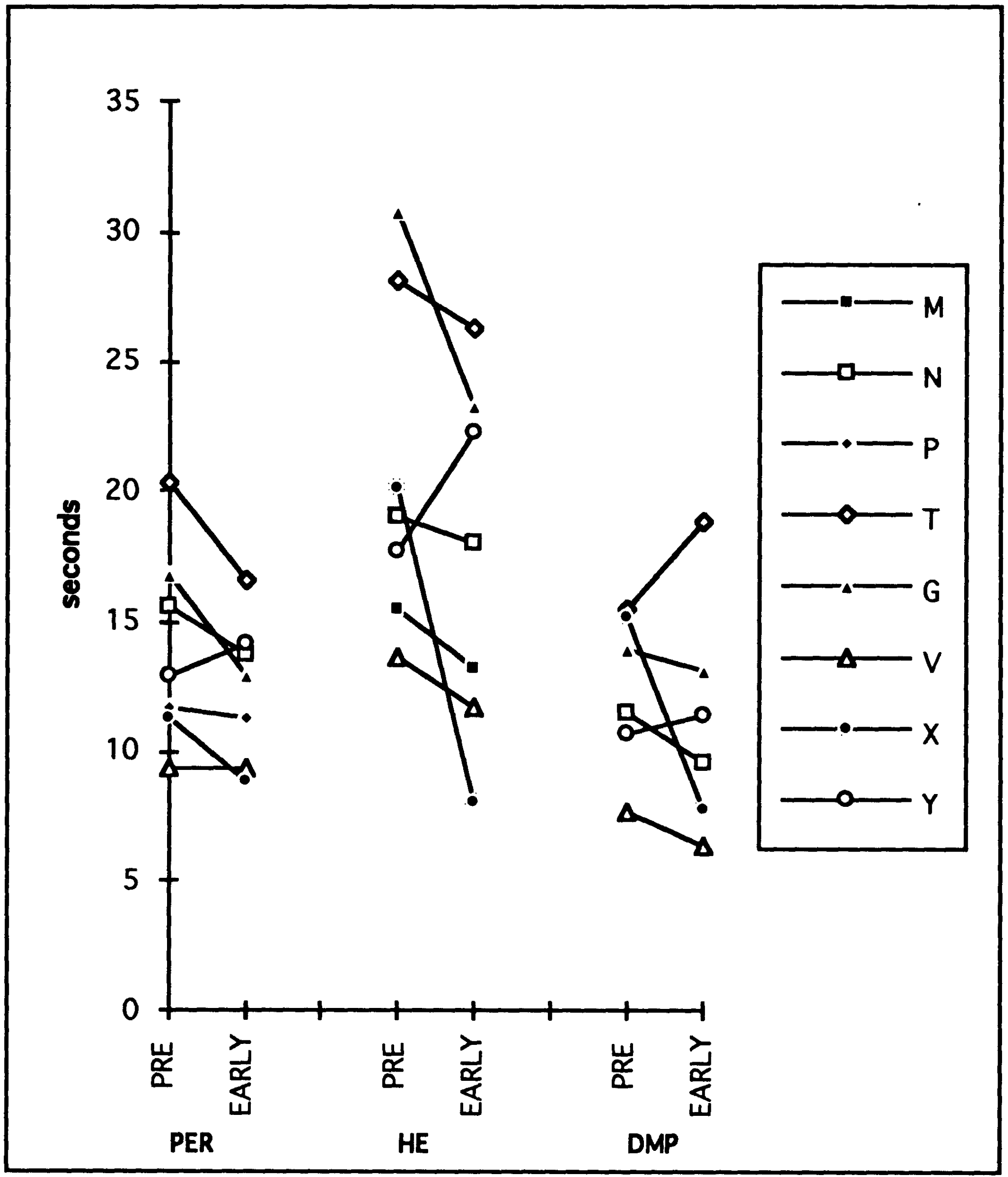


Figure 5. Changes in mean time constant for per- and post-rotatory (head-erect and dumping) segments between pre- and early postflight testing sessions

Table 6. t statistics for τ comparisons between pre- and early postflight testing

	PER			HE			DMP		
SUBJ	t	dof	p≤	t	dof	p≤	t	dof	p≤
G	-	-	-	3.415	3.4	0.034	0.639	9.4	0.538
N	3.849	28.1	0.001	1.701	20.0	0.104	1.949	8.0	0.087
T	2.260	24.4	0.033	0.835	6.1	0.435	-1.074	3.5	0.352
X	2.777	8.6	0.023	-	-	-	-	-	-
M	-	-	-	-	-	-	-	-	-
P	0.423	14.7	0.678	-	-	-	-	-	-
V	-0.084	21.0	0.934	2.187	18.0	0.042	1.647	11.8	0.126
Y	-1.132	16.8	0.273	-2.900	8.6	0.018	-0.980	14.9	0.343

PER, per-rotatory; HE, post-rotatory head-erect; DMP, post-rotatory dumping maneuver, - not enough data for a t-test; results for which p≤0.05 are underlined

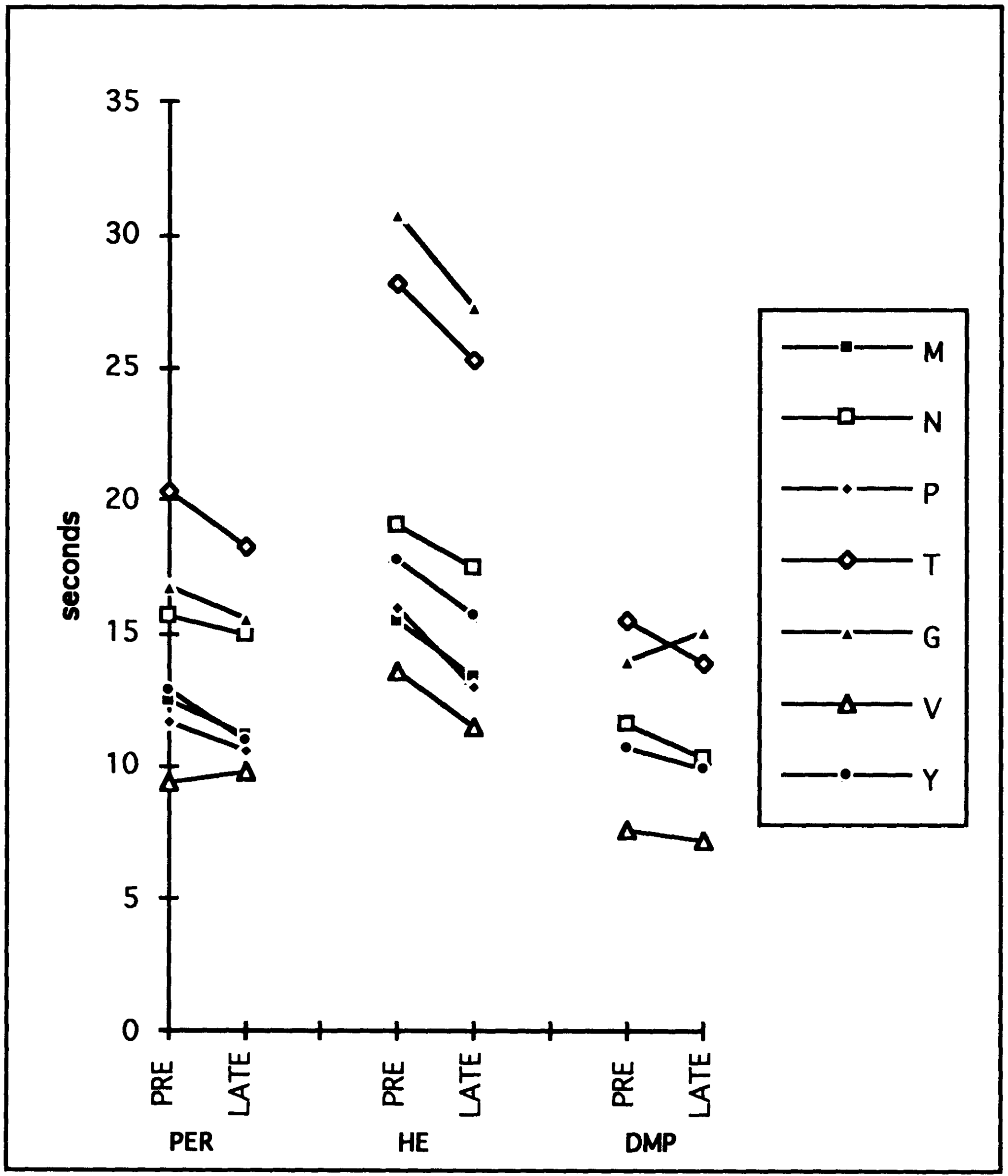


Figure 6. Changes in mean time constant for per- and post-rotatory (head-erect and dumping) segments between pre- and late postflight testing sessions

Table 7. t statistics for τ comparisons between pre- and late postflight testing

	PER			HE			DMP		
SUBJ	t	dof	p≤	t	dof	p≤	t	dof	p≤
G	-	-	-	0.634	2.0	0.589	-1.359	12.1	0.199
N	1.193	27.1	0.243	1.350	9.5	0.208	1.182	11.3	0.261
T	0.753	5.9	0.481	0.779	1.7	0.528	1.068	10.6	0.309
X	-	-	-	-	-	-	-	-	-
M	1.890	11.1	0.085	1.273	3.7	0.278	-	-	-
P	1.159	8.2	0.279	4.579	6.3	0.003	-	-	-
V	-1.126	36.5	0.268	2.279	17.4	0.036	0.487	13.6	0.634
Y	2.185	27.9	0.037	0.699	3.3	0.530	1.042	17.3	0.312

PER, per-rotatory; HE, post-rotatory head-erect; DMP, post-rotatory dumping maneuver, - not enough data for a t-test; results for which $p \leq 0.05$ are underlined

Dumping Results:

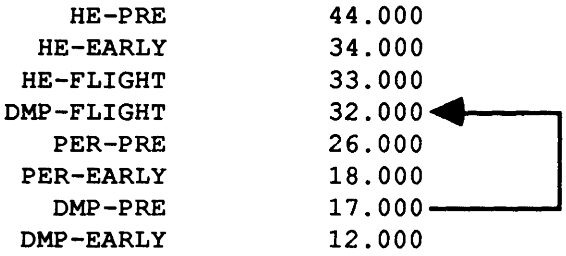
On earth (1-G), the dumping maneuver reduces τ relative to head-erect controls (Oman and Balkwill 1993). Figure 8 confirms this for 1-G, but paints a different picture of dumping in μG. As expected, within-subject comparisons between HE and DMP show these decreases systematically during ground sessions (PRE, EARLY, LATE), but not in μG. In fact, three out of seven of the subjects have τ increases inflight, and the decreases in the others are not nearly as substantial as they are in (1-G) ground tests.

Comparisons between preflight and inflight means (for each subject and segment type) indicate (Figure 4) that subjects' dumping time constants generally increase inflight relative to preflight controls.

A Friedman rank sum test (McClave 1988) was used to determine if there was a systematic tendency for the group of subjects to have systematically higher τ's for some condition x segments cells = blocks (e.g. PER-PRE...DMP-LATE) than others. The sum of subject ranks for each block is the basis for the test statistic, which is compared to a χ^2 distribution.

Subjects M and P could not be included in this test because of missing data. The per-rotatory inflight condition and all 3 late postflight conditions were also omitted (as blocks) for the same reason.

The Friedman test statistic, 22.4, rejected the null hypothesis (that rank sums are equal) with $p=0.002$ (df=7). Subjects ranked inflight dumping consistently higher than preflight dumping (Figure 7) ($p<0.05$). (The arrow in Figure 7 shows how much higher inflight dumping, on average, is ranked over preflight dumping.) Other interesting comparisons, such as those between preflight per-rotatory blocks and early postflight per-rotatory blocks, showed no significant results.



HE-PRE	44.000
HE-EARLY	34.000
HE-FLIGHT	33.000
DMP-FLIGHT	32.000
PER-PRE	26.000
PER-EARLY	18.000
DMP-PRE	17.000
DMP-EARLY	12.000

Figure 7. Ordered Friedman rank sums. Arrow indicates comparison between preflight and inflight dumping segments

A univariate repeated measures (3x3) (PER, HE, and DMP)x(PRE, FLIGHT, and EARLY)=(segment x condition) ANOVA of means of subjects N, G, V, X, and Y was performed. This test of within-subject differences, found significant segment type ($F[2,8]=19.702$; $p=0.001$) and segment x session type interaction effects ($F[4,16]=3.564, p=0.029$). Condition effects were not statistically significant. (Other subjects were not included due to missing data.)

Two contrasts were tested and gave significant results. One tested if the effect of flight (as against averaged ground conditions) was different for DMP as against head-upright segments. (i.e. testing the hypothesis: $\left[\frac{\tau_{PER}+\tau_{HE}}{2}-\tau_{DMP}\right]_{PRE}-\left[\frac{\tau_{PER}+\tau_{HE}}{2}-\tau_{DMP}\right]_{FLIGHT}=0$). It is found that the excess of the mean of (PER + HE) over DMP is significantly greater for PRE than for FLIGHT conditions, ($F[1,4]=14.414$; $p=0.019$).

Another contrast tested if the increase in post-rotatory head-erect over dumping segments was different inflight from what it was preflight. (i.e. testing the hypothesis:

$\left[\frac{\tau_{PRE}+\tau_{EARLY}}{2}-\tau_{FLIGHT}\right]_{HE}-\left[\frac{\tau_{PRE}+\tau_{EARLY}}{2}-\tau_{FLIGHT}\right]_{DMP}=0$), (F[1,4]=33.421; p=0.004), i.e., the excess of the mean of (PRE+EARLY) over FLIGHT is greater for HE than for DMP.

As individuals, only Group I subjects G and X's time constants decreased between preflight and inflight testing sessions (but not significantly by independent sample t-tests). (This result might have been expected because Group I subjects show an inflight VS reduction anyhow.) Subject X was one of the few subjects who did not show significant dumping time constant decreases from head-erect (by independent sample t-tests) during preflight control sessions.

Table 9 shows that none of the subjects showed significant inflight dumping reductions. Such reductions might not have been expected of Group I subjects, who had diminished inflight VS. However, there is no indication that either of the Group II subjects, who had presumably adapted VS, had dumping reductions either. This is particularly interesting, since if the dumping mechanism had been present inflight, they would have shown the same dumping decreases observed during ground tests.

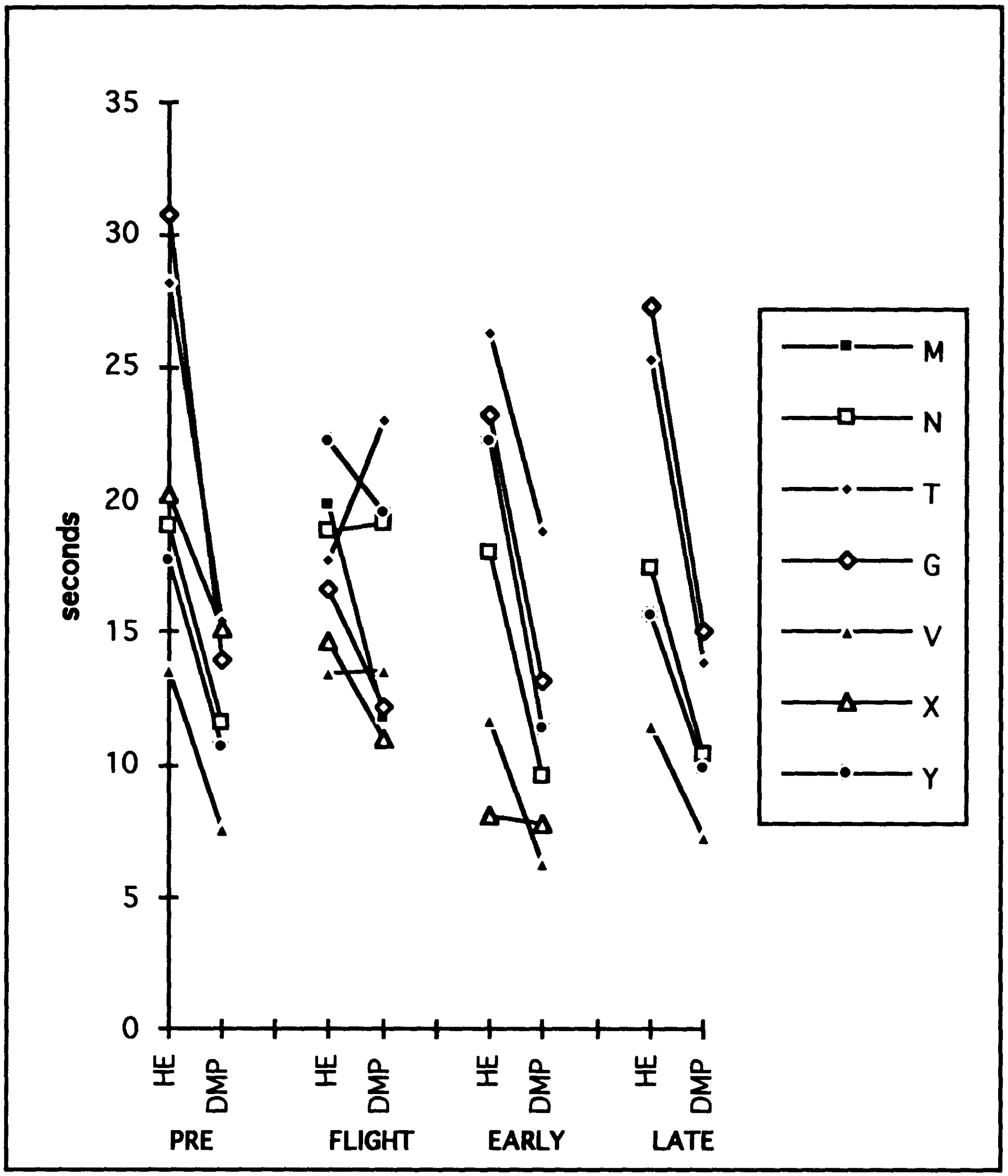


Figure 8. Changes in mean time constant during preflight, inflight, and postflight (both early and late sessions) between post-rotatatory head-erect and dumping segments

Table 8. t statistics for τ comparisons between post-rotatory (head-erect) and dumping maneuver testing

	PRE			FLIGHT			EARLY			LATE		
SUBJ	t	dof	p≤	t	dof	p≤	t	dof	p≤	t	dof	p≤
G	21.724	2.8	0.000	0.719	2.3	0.539	4.131	5.0	0.009	2.261	2.1	0.149
N	10.388	25.1	0.000	-	-	-	9.412	5.5	0.000	5.146	12.6	0.000
T	6.286	12.5	0.000	-	-	-	2.240	3.8	0.092	3.316	1.3	0.134
X	1.173	6.7	0.281	1.149	10.8	0.275	-	-	-	-	-	-
M	-	-	-	-	-	-	-	-	-	-	-	-
P	-	-	-	-	-	-	-	-	-	-	-	-
V	6.794	17.9	0.000	-0.071	9.0	0.945	6.889	11.7	0.000	5.254	12.9	0.000
Y	8.145	25.0	0.000	1.221	4.7	0.279	7.192	7.4	0.000	1.945	3.2	0.141

PRE, preflight; FLIGHT, inflight; EARLY, early postflight; LATE, late postflight, - not enough data for a t-test; results for which $p \leq 0.05$ are in underlined

Effect of Gravity on Head-Erect Per- and Post-rotatory Differences in τ:

It is well-established that because of neural adaptation, the post-rotatory τ exceeds the corresponding per-rotatatory value for head-erect runs. It is widely believed that gravity effects would change these two time constants in equal measure. To confirm this, a possible gravity effect was sought among the various session types (PRE, FLIGHT, EARLY, LATE):

Available pair differences were averaged within runs ($\tau_{diff}=\tau_{HE}-\tau_{PER}$). Numbers of τ_{diff} measurements by subject and condition is given in Table 9 and their means are shown in Figure 9. This figure does not suggest any distinctive trends in τ_{diff} among the various testing conditions.

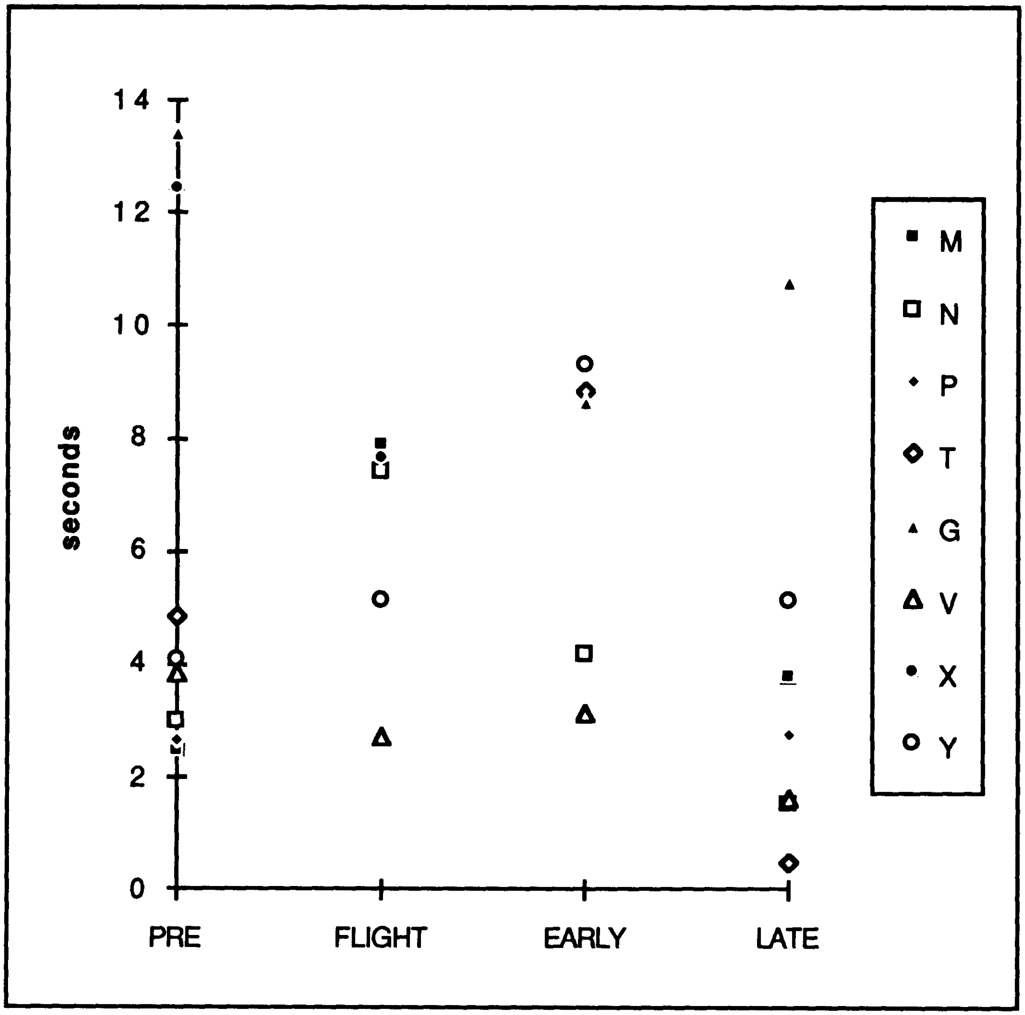


Figure 9. τ_{diff}, the difference between the post-rotatory (head-erect) and per-rotatory time constant within a given run, averaged within subjects and testing conditions

Table 9. Number of available τ_{diff} data points within each subject and experimental condition

SUBJ	PRE	FLIGHT	EARLY	LATE
M	3	1	0	2
N	14	1	8	8
P	3	0	0	1
T	5	0	2	1
G	1	0	2	3
V	11	1	5	6
X	3	4	0	0
Y	12	2	6	2

PRE, preflight; FLIGHT, inflight; EARLY, early postflight; LATE, late postflight

Engineering Results

Changes to the implementation of signal processing methods from the previous analysis (Pouliot 1995) (calculation of SPV from raw data and further preparation for parameter identification) were necessitated by the change of software (MATLAB 3.0 to MATLAB 4.0). However, these changes had the beneficial effect of decreasing the amount of time the software semi-automatically analyzed a session of data: During the previous analysis, approximately 6-7 hours of computer time was spent analyzing one session of data (Pouliot 1995). For the present analysis, one session of data was analyzed in approximately two hours. The run segmentation/decimation program (which iteratively decimated and filtered time series of high sampling rates) used in the previous analysis was particularly time consuming, spending approximately one hour of computer time per subject per session. The current program (which uses MATLAB automatic resampling function) spent approximately fifteen minutes per subject per session.

Another improvement to the "analysis pipeline" (Pouliot 1995) consisted of a program, incorporated in the semi-automatic session analysis software, which performed segment outlier detection and placed remaining time constant and gain data, identified by session type, run segment type (PER, HE, DMP), experimental condition type (PRE, FLIGHT, EARLY, LATE), session number, and run number, into a tabulated text file. This text file could be easily imported into various other software packages such as SYSTAT or EXCEL, saving the investigator from spending time manually copying data for statistical analyses.

Discussion

Pooling the results from all SLS-1 and SLS-2 subjects in this study, and considering them as a single group, it was not possible to demonstrate a significant change in inflight head-upright time constant. Although this implies that the VS of subjects, on average, adapted to preflight levels by the time orbital testing sessions occurred, closer inspection of the data confirmed a conclusion of SLS-2 that different subjects adapted differently to µG (Oman, Pouliot et al. 1996). From inflight data, subjects could be categorized as "slow adapters" (Group I) and "fast adapters" (Group II).

The previous SLS-2 analysis, however, also concluded that one's being a "slow adapter" was correlated with previous flight experience (Oman, Pouliot et al. 1996). This study, which used a more limited set of subjects from the SLS-1 and SLS-2 missions (the SLS-2 study also included results from the IML-1 analysis), did not find this same correlation. Of the seven individuals who participated in the study, one could a priori predict classification via flight experience for only five. However, a clearer picture of this correlation would improve with a larger group of subjects. Thus, it would be interesting to perform a reanalysis of the other three missions (using present analysis techniques) to determine whether the groupings of other subjects could be determined from previous experience.

The effect of readaptation to earth's gravity on the long head-erect VOR time constant was somewhat clarified in this study. A previous hypothesis stated that this time constant should be reduced after return to earth due to relatively unfamiliar otolith cues (Oman, Pouliot et al. 1996). However, this study found that for the group of subjects, early reductions are more subtle than those from dumping or parabolic flight effects. It is interesting, however, that reductions existed more for Group I subjects than for Group II subjects. One could say that since Group II subjects quickly adapted inflight, that they also quickly adapted postflight such that by the time early postflight sessions occurred, head-upright time constants were at preflight levels. However, the data could also suggest that Group I subjects never regained preflight τ's and that their decreased time constants observed postflight were residual effects.

Furthermore, subject Y's per-rotatory and post-rotatory head-erect mean time constants actually increased in early postflight testing (relative to preflight controls) by respectively 110% and 126%. These increases resemble Y's increases observed between preflight and inflight. In addition, time constant means for landing day measurements of subject Y (per-rotatory: 14.7 s, post-rotatory head-erect: 22.9 s) were comparable to lengthened (with respect to preflight) average measurements inflight, suggesting that this subject's increased VS persisted in early postflight tests. It was not until late postflight did this subject's time constants decrease to comparable preflight values.

One could speculate that during the period immediately after return to the 1-G environment, Y kept an increased VS that had been acquired in space, and only with further exposure to earth's gravity and everyday motions and head movements did Y's time constants return to preflight values. Previous studies have shown that during a period immediately following exposure to μG, head-erect τ's will decrease to values comparable with those observed in dumping experiments (DiZio and Lackner 1988; Oman, Pouliot et al. 1996). It has also been hypothesized that subjects who have adapted in μG would demonstrate similar decreases after return to earth. However, significant decreases in head-erect time constants between preflight and early postflight sessions were not found for Group II subjects.

It was originally hypothesized that by late postflight, head-erect time constants would have been restored to their respective preflight values. The analysis demonstrated, however, that for all subjects, there was a slight decrease in the means of these late postflight time constants relative to the preflight controls. One might say that the CNS had not quite finished its task of returning to preflight VS. (Recall that the latest postflight session for either mission was within two weeks after the landing day.) However, another plausible explanation for this observation is that the subjects, who were not physically active in the weeks after return, were not making the same kinds of head movements as they did preflight.

It was confirmed in this study, to a higher degree of certainty, that the dumping maneuver, performed after subjects have adapted in μG, does not reduce time constants as it does in preflight controls. This is in part confirmed by the statistical insignificance of differences between head-erect and dumping inflight time constant means. It is also demonstrated by significant increases in dumping time constant means from preflight controls to inflight sessions. Further, changes between preflight and inflight dumping time constants were shown to be significantly smaller than corresponding average head-upright time constant changes.

Inflight dumping decreases would not be expected, since the otolith signal changes implicated in causing these decreases should not occur in a μG environment. It was hypothesized, however, that dumping time constant decreases, if observed during inflight tests, could be attributed to haptic cues. However, no subjects could be found in this study who demonstrated such haptic dumping of VS inflight at a statistically significant level. Although subjects X and G did show a decrease in dumping time constant, neither change was statistically significant. If there actually was a haptic dumping effect in these subjects, the effect should be small, because both subjects head upright time constants were relatively short (Group I), even without head tilt. Haptic dumping would have produced only a small further decrease. In addition, as determined both in this study and the SLS-2 analysis (Oman, Pouliot et al. 1996), the dumping maneuver's effect on subject X, during preflight trials, did not have a large effect in terms decreasing the time constant relative to when the head was erect.

Suggestions For Further Research

Data acquisition and processing software have improved such that time series analyses can be performed in a near real-time manner. Investigators should thus take advantage of this software during experimental sessions such that runs can be re-performed if the data quality forces exclusions.

For orbital tests, a test session on the final day inflight would be useful in terms of determining the degree of VS adaptation gained over the course of the flight; particularly for subjects who are thought to adapt more slowly.

Likewise, an effort should be made to ensure ground tests for all subjects on the landing day, so that the acute effect of transition from the μG environment to earth's gravity can be truly investigated. Another possibility for future investigations is to concentrate postflight testing sessions on individual days, so as not to blur temporal resolution by averaging data collected within 3 day periods.

Conclusions

The combined data of the SLS-1 and SLS-2 missions has offered the unique opportunity to study the effect of weightlessness on the VOR of a larger group of subjects than ever possible. Findings of previous rotating chair experiments have been confirmed while some earlier hypotheses have been slightly modified.

It was determined in this study, as in the previous analysis, that subjects do not adapt to weightlessness equally. Instead, subjects can generally be classified as being either slow (Group I) or fast (Group II) adapters to μG, based on whether or not inflight time constants are less than corresponding preflight controls. It should be reiterated that this classification does not imply that subjects within a given classification adapt equally. μG adaptation more realistically would be represented by a continuum of possible responses (Oman, Pouliot et al. 1996).

Furthermore, the groupings of subjects in the previous SLS-2 analysis was confirmed by this study. In addition, the subject who participated in both missions was grouped in the same category. However, a correlation between a subject's classification and previous flight experience was not confirmed by the present study, possibly because of the smaller number of subjects considered.

Table 10 summarizes the head-upright τ changes (from preflight controls) for all subjects and for sub-groups (I, II).

Table 10. τ changes between preflight and non-control sessions

		PER			HE		
	n	FLIGHT	EARLY	LATE	FLIGHT	EARLY	LATE
All	8	-0.02 s (-0.2%)	-1.0 s (-3.9%)	-1.1 s (-8.0%)	-3.1 s (-14.9%)	-3.2 s (-15.2%)	-2.5 s (-12.2%)
Group I	4	-3.8 s (-26.0%)	-3.0 s (-18.6%)	-1.3 s (-7.5%)	-7.6 s (-30.9%)	-5.6 s (-22.9%)	-2.6 s (-10.2%)
Group II	4	2.8 s (24.1%)	0.32 s (2.9%)	-0.99 s (-8.5%)	2.9 s (18.6%)	0.15 s (0.94%)	-2.3 s (-14.8%)

PER, per-rotatory; HE, post-rotatory head-erect; FLIGHT, inflight; EARLY, early postflight; LATE, late postflight

Decreases between head-erect preflight and early postflight time constants were confirmed only for Group I subjects. One Group II subject (Y), in fact, had a slight increase in τ relative to preflight controls. This suggests a possibility that early postflight decreases in τ were more a residual effect of inflight decreases than an effect of unfamiliar otolith cues upon reentry.

Decreases between head-erect late postflight time constants and corresponding preflight controls were observed in all subjects. These decreases suggest that for the group of subjects, time constants did not recover to preflight values by late postflight testing.

Significant inflight dumping time constant increases were found for the group of seven subjects (relative to ground tests: FLIGHT/PRE=1.3; FLIGHT/EARLY=1.5). Furthermore, a slight dumping decrease observed inflight (DMP/HE=0.89) was not statistically significant.

References

Balkwill, M. D. (1992). Changes in human horizontal angular VOR after the Spacelab SLS-1 mission, Massachusetts Institute of Technology.

DiZio, P. and J. R. Lackner (1988). "The effects of gravitoinertial force level and head movements on post-rotational nystagmus and illusory after-rotation." Experimental Brain Research **70**: 485-495.

Engelken, E. J. and K. W. Stevens (1990). "A new approach to the analysis of nystagmus: an application for order-statistic filters." Aviation, Space, and Environmental Medicine. **61**: 859-864.

Gonshor, A. and G. M. Jones (1971). "Extreme vestibulo-ocular adaptation induced by prolonged optical reversal of vision." Journal of Physiology London **256**: 381-414.

Grace, A. (1990). Optimization Toolbox for MATLAB. South Natick, MA, The Mathworks.

Liefield, T. (1993). Changes in human horizontal angular VOR after the Spacelab SL-1 mission, Massachusetts Institute of Technology.

Massoumnia, M. (1983). Detection of fast phase nystagmus using digital filtering., Massachusetts Institute of Technology.

McClave (1988). Statistics. San Francisco, Dellen.

Oman, C. M. and M. D. Balkwill (1993). "Horizontal angular VOR, nystagmus dumping, and sensation duration in Spacelab SLS-1 crew members." Journal of Vestibular Research **3**: 315-330.

Oman, C. M. and D. S. Calkins (1993). Effect of orbital flight on the human horizontal angular vestibulo-ocular reflex response to 120 deg/sec step stimuli, NASA Johnson Space Center Neurophysiology Laboratory.

Oman, C. M. and M. J. Kulbaski (1988). "Spaceflight affects the 1-G postrotatory vestibulo-ocular reflex." Advanced Oto-Rhino-Laryngology **42**: 5-8.

Oman, C. M., C. F. Pouliot, et al. (1996). "Horizontal angular VOR changes in orbital and parabolic flight: human neurovestibular studies on SLS-2." Journal of Applied Physiology **81**(1): 69-81.

Oman, C. M. and H. J. Weigl (1989). Postflight vestibulo-ocular reflex changes in space shuttle/Spacelab D-1 crew. Aerospace Medical Association Annual Meeting, Washington, DC.

Pouliot, C. F. (1995). Changes in the horizontal angular vestibulo-ocular reflex of SLS-2 astronauts due to weightlessness, Massachusetts Institute of Technology.

Raphan, T., V. Matsuo, et al. (1979). "Velocity storage in the vestibulo-ocular reflex arc (VOR)." Experimental Brain Research **35**: 229-248.

Robinson, D. A. (1977). Vestibular and optokinetic symbiosis: an example of explaining by modeling. Control of Gaze by Brain Stem Neurons. R. Baker and A. Berthoz. Abbaye de Royaumont, Paris, France, Elsevier/North-Holland Biomedical Press. **1**: 49-57.

Steinhausen, W. (1933). "Über die Beobachtungen der Cupula in den Bogengangsumpullen des Labyrinthes des lebenden Hechts." Pflügers Arch **232**: 500-512.

Van Egmond, A. A. J., J. J. Groen, et al. (1949). "The mechanics of the semicircular canals." Journal of Physiology London **110**(1-17).

Appendix - MATLAB and C Scripts

This appendix includes analysis scripts the author has written or to which the author has made major modifications:

AATM.c (C source code for AATM.mex)
Batch_Analyse
CODES
Ind_Model_Fit
Multiple_AATM
OSfilt_Diff_AATM
PFMH.c (C source code for OSPFMH.mex)
Run_Report_Maker
Save_Dat_To_File
Stat_Prep_Batch

These codes were specific to the format and identification of SLS-1 ground data. Scripts for inflight and SLS-2 data were slightly modified and are not included.

```
/*AATM.c*/
/*Code For mex interface script, aatm.mex*/
#include <stdio.h>
#include <math.h>
#include "mex.h"

#ifndef DOUBLE
#ifdef THINK_C      /* THINK C doubles are extended, ints are short */
#define DOUBLE            double /* need true double for MATLAB matrices */
#define INT               int                /* need long for user_fcn() */
#else               /* everyone else does the normal thing */
#define DOUBLE            double
#define INT               int
#endif
#endif

/* Input Arguments */

#define      LENGTH_IN    prhs[0]
#define      VEL_IN       prhs[1]


/* Output Arguments */

#define      SPV_OUT            plhs[0]


#define      max(A,B)     ((A) > (B) ? (A) : (B))


#define      min(A,B)     ((A) < (B) ? (A) : (B))


void mexFunction(    /* Matlab mex-file interface routine */
       int nlhs,
       Matrix *plhs[],
       int nrhs,
       Matrix *prhs[]
       )
{
       void AATM();
       double *length, *vel, *spv;
       int m, n, num_samples;

       /* Check for proper number of arguments */
       if (nrhs != 2) {
             mexErrMsgTxt("AATM requires two input arguments.");
             }
       else if (nlhs != 1) {
             mexErrMsgTxt("AATM requires one output argument.");
             }

       /* Ensure that LENGTH is a scalar, and VEL a vector. */
       m = mxGetM(LENGTH_IN);
       n = mxGetN(LENGTH_IN);
       if ((m != 1) || (n != 1)) {
             mexErrMsgTxt("AATM requires that LENGTH be an integer scalar.");
             }
       m = mxGetM(VEL_IN);
       n = mxGetN(VEL_IN);
       if ((max(m,n) <= 1) || (min(m,n) != 1)) {
             mexErrMsgTxt("AATM requires that VEL be a vector.");
             }
```

```
        /* Create a matrix for the return argument */
        SPV_OUT = mxCreateFull(m, n, REAL);

        /* Assign pointers to the various parameters */
        spv = mxGetPr(SPV_OUT);
        length = mxGetPr(LENGTH_IN);
        vel = mxGetPr(VEL_IN);

        /* Do the actual computations in a subroutine */
        num_samples = max(m,n);
        AATM( length, num_samples, vel, spv );

}


#define ALPHA 0.44
#define BETA  0.12
#define      MU           0.4

void AATM(
       double *length,
       int num_samples,
       double vel[],
       double spv[]
       )
{
    void bubble_sort();
    int find_index(), check_sort();
    int i, j, k, k1, k2;
    int n, stop;
    int N, L, K;
    int Lalpha, Lbeta, M;
    double Sbeta, sum;
    double *s;
    double old, new;

    /* integer number of samples in sliding window */
    N = floor(*length / 2);
    L = 2 * N + 1;
    stop = num_samples - N;

    /* initialize skewing parameters */
    Lalpha = L * ALPHA;
    Lbeta = L * BETA;
    M = L * MU;

    /* set up array for storing sorted window */
       s = (double *)mxCalloc( L, sizeof(double) );
/*     s = create_real_array(L);*/

    /* initialize and sort first window of data */
    for (i = L-1; i >= 0; i--)
       s[i] = vel[i];
    bubble_sort(s,L);

if (!check_sort(s,L)) {
       printf("Unsorted list at index %ld.\r",n);
    for (i = 0; i < L; i++)
        printf("%f : ",(float)s[i]);
    }

    /* calculate skewness of first window of data */
    if (s[L - Lbeta - 1] == s[Lbeta - 1])
       Sbeta = 0;
```

```
    else
       Sbeta = (s[L - Lbeta - 1] + s[Lbeta - 1] - 2 * s[N]) / (s[L - Lbeta - 1] -
s[Lbeta - 1]);
    K = - Sbeta * M;

    /* new value at centre of window is mean of estimated peak of histogram */
    sum = 0;
    for (i = (Lalpha + K); i < (L - Lalpha + K); i++)
       sum = sum + s[i];
    spv[N] = sum / (L - 2 * Lalpha);

    /* for each value */
    for (n = N + 1; n < stop; n++) {

if (!check_sort(s,L)) {
       printf("Unsorted list at index %ld.\r",n);
/*     for (i = 0; i < L; i++)
         printf("%f : ",(float)s[i]);*/
    }


       old = vel[n - N - 1];        /* value to be removed from list */
       new = vel[n + N];            /* value to be inserted into list */

       k1 = find_index( s, old, 0, L - 1 );     /* find index of old value */

       if (old == new) {    /* simple replacement */

             s[k1] = new;
             spv[n] = spv[n - 1];/* histogram has not changed, so SPV has not
either */

             }
       else {

             /* check for out of bounds of sorted list */
             if (new <= s[0])
                    k2 = 0;
             else if (new >= s[L - 1])
                    k2 = L;
             else if (new > old)
                           k2 = find_index( s, new, k1, L - 1 );    /* find
insertion index for new value */
             else
                           k2 = find_index( s, new, 0, k1 ); /* find insertion
index for new value */


/*   printf("n, k1, k2 = %ld, %ld, %ld\r",n,k1,k2);*/
                    /* shift list, if necessary, to remove old value */
             if (k1 < k2) {
                    k2--;
                    for (k = k1; k < k2; k++)
                           s[k] = s[k+1];
                    }
             else if (k1 > k2) {
                    for (k = k1; k > k2; k--)
                           s[k] = s[k-1];
                    }

             /* insert new value into list */
             s[k2] = new;

                 /* calculate skewness of new window of data */
```

```
                if (s[L - Lbeta - 1] == s[Lbeta - 1])
                        Sbeta = 0;
                else
                        Sbeta = (s[L - Lbeta - 1] + s[Lbeta - 1] - 2 * s[N]) / (s[L -
Lbeta - 1] - s[Lbeta - 1]);
                K = - Sbeta * M;

                        /* new value at centre of window is mean of estimated peak of
histogram */
                sum = 0;
                for (i = (Lalpha + K); i < (L - Lalpha + K); i++)
                        sum = sum + s[i];
                spv[n] = sum / (L - 2 * Lalpha);

                }
        }


        for (i = N; i > 0; i--) {
                spv[i - 1] = vel[i - 1];
                spv[num_samples - i] = vel[num_samples - i];
                }

    mxFree(s);
}


void bubble_sort(
        double *x,
        int n
        )
{
        int i, j;
        double mx;
        int index;

        for (i = n - 1; i > 0; i--) {
                index = i;
                mx = x[index];
                for (j = i - 1; j >= 0; j--) {
                    if (x[j] > x[index]) {
                                index = j;
                                mx = x[j];
                                }
                        }
                x[index] = x[i];
                x[i] = mx;
                }
}


int find_index(
        double list[],
        double value,
        int i,
        int j
        )

{
        int k;

        k = floor((i + j) / 2);
        while (i != k) {    /* list[i] <= value */
                if (list[k] == value)
```

```
                        break;
                else if (list[k] < value)  /* move left boundary */
                        i = k;
                else                                    /* move right boundary */
                        j = k;
                k = floor((i + j) / 2);
                }

        if (list[k] != value)
                k++;

        return(k);
}


int check_sort(
        double x[],
        int n
        )

{
        int i;
        int sorted;
        int dummy;

        dummy=x[0];
        sorted = TRUE;       /* assume OK */
        for (i = n - 1; i > 0; i--) {
                if (x[i] < x[i-1]) {
                        sorted = FALSE;
                        break;
                        }
                }
        return(sorted);
}
```

```
% batch_analyse

% written by T. Liefeld throughout spring 93
% given a folder of runs from a BDC,  this functions as a
% superscript that will prompt the user for all analysis
% from data collection through to model fitting.
%
% slight organizational organizational modifacations by C. Pouliot 10/94
% and M. Neimark 6/96

clear
hold off

hard_disk = 'HD:';


data_path=input('Enter Data Path (include end colon)>> ','s');
sub_code =input('Enter Subject Code                  >> ','s');
number =  input('Enter Number of Runs                >> ');

% Generate places to store the time constant and gain data.
tc_vect=zeros(2,number);
gain_vect=zeros(2,number);
pgood_vect=zeros(2,number);
MSE_vect=zeros(2,number);

% For each run, there is one per-rot, and one post-rot.

tc_vect=tc_vect*NaN;
gain_vect=gain_vect*NaN;
pgood_vect=pgood_vect*NaN;
MSE_vect=MSE_vect*NaN;

q = input('Calibration factors from file or new? (f/n) >> ','s');
if ((q == 'f') | (q == 'F'))
      cal_from_file
end
if ((q == 'n') | (q == 'N'))
      cal_factor_gen
end

flag=1;
run_problem=zeros(1,number);
qq=input('Would you like to designate abnormal runs for this session? >> ','s');
if ((qq == 'y') | (qq == 'Y'))
      fprintf(1,'Enter Run # (do not include cals) and problem code.\n');
      fprintf(1,'To quit type "0" after run prompt.\n');
      while flag==1,
            runnum=input('Run >>');
            if runnum==0,
                  break;
            end
            probcode=input('Problem Code >>');
            run_problem(runnum)=probcode;
      end
      qq=input('Save Problem Codes? >> ','s');
      if ((qq == 'y') | (qq == 'Y'))
            eval(['save ',data_path,sub_code,'.prob run_problem']);
      end
else
      qq=input('Would you like to load other problem codes? >>','s');
      if ((qq == 'y') | (qq == 'Y'))
            eval(['load ',data_path,sub_code,'.prob -mat']);
      end
```

```
end

q2 = input('Do you want to perform AATM? >> ','s');
if ((q2 == 'y') | (q2 == 'Y'))
    multiple_AATM;
end

% create the run_code matrix, codes
CODES
n_codes=length(codes(:,1));
number = n_codes;

q3 = input('Do you want to perform Tachan >> ','s');

j=1; % Problem code index
if ((q3 == 'y') | (q3 == 'Y'))
    for i = 1:number
             j=j+1;
             rp=run_problem(j);
             if (j==6|rp==1|rp==4|rp==6),
                    j=j+1;
             end
        run_code = codes(i,:);
        fprintf(['\nRun code = ',run_code,'\n']);
             if run_problem(j)==2,
                    tachan_batchpost;
             elseif run_problem(j)==3,
                    tachan_batchper;
             else
             tachan_batch;
             end
    end
end

q4 = input('Do you want to perform stat prep >> ','s');

if ((q4 == 'y') | (q4 == 'Y'))
    for i = 1:number
        run_code = codes(i,:);
        fprintf(['\nRun code = ',run_code,'\n']);
        stat_prep_batch;
    end
end

q5 = input('Do you want to fit a Model? >> ','s');
if ((q5=='y') | (q5=='Y')),
       j=1;  %See Explanation of j in ind_model_fit_exp
       for i = 1:number,
             run_code = codes(i,:);
       ind_model_fit_exp;
       end
       eval(['save ',data_path,sub_code,'.gtc',' tc_vect gain_vect MSE_vect'])
end

q6=input('Do you want to save run-segment percent good data?>>','s');
if ((q6=='y') | (q6=='Y')),
       j=1;
       for i = 1:number
             run_code = codes(i,:);
             eval(['load ',run_code,'.dec_good1 -mat']);
             j=j+1;  % This is a run counter which indexes runs according
                           % to their actual protocolled number.  Note that
                           % i is an index only for non-calibrated runs.
             rp=run_problem(j);
```

```
                if (j==6|rp==1|rp==4|rp==6),
                        j=j+1;  % Removes run number 6 which is a cal run or no data
                end                % runs.  NOTE:THIS CODE IS SPECIFIC FOR THE SLS-1
                                                 PROTOCOL.
        %Only first 25 seconds (100 samples) are checked
                %for percent of good data
        pgood_vect(1,j)=sum(dec_good(1:100));
                pgood_vect(2,j)=sum(dec_good(241:340));
        end
        eval(['save ',data_path,sub_code,'.pg',' pgood_vect']);
end

eval(['chdir ',hard_disk]);
q7=input('Do you want to add this data to analysis?>>','s');
if ((q7=='y') | (q7=='Y')),
        q=input('Which File would you like to store the data in?>>>','s');
        if exist(q) ~= 2
                eval(['load ',data_path,sub_code,'.gtc -mat'])
                eval(['load ',data_path,sub_code,'.pg -mat'])
                fid=fopen(q,'w');
                fprintf(fid,'SUBJ SEG COND DIR TC GAIN RUN SSN FB\n');
                sav_dat_to_file;
                fclose(fid);
        else
                eval(['load ',data_path,sub_code,'.gtc -mat'])
                eval(['load ',data_path,sub_code,'.pg -mat'])
                fid=fopen(q,'a');
                sav_dat_to_file;
                fclose(fid);
        end
end

q8=input('Do You Want to Generate a Run Report for this session?>>','s');
if ((q8=='y') | (q8=='Y')),
        eval(['chdir ',hard_disk,'reports']);
        fid=fopen([sub_code,'.rep'],'w');
        eval(['load ',data_path,sub_code,'.gtc -mat'])
        eval(['load ',data_path,sub_code,'.pg -mat'])
        run_report_maker;
        fclose(fid);
end
```

```
% CODES
%
% written by T. Liefeld, 21/4/93
% modified by M. Neimark 7/96
% creates a matrix, called codes,  containing all the run codes
% of non-calibration runs snd runs actually performed.

j = 0;
for i = 1:number
    rp=run_problem(i);
    if (run(i) == 1 & (rp~=1 & rp~=4 & rp~=6))
        stln = length([data_path,sub_code])+2;
    if (i < 10)
        n = [num2str(0),num2str(i)];
    else
        n = num2str(i);
    end
    codes(i-j,1:stln) = [data_path,sub_code,n];
        elseif (run(i) == 0)
        j = j+1;
    end
end
```

```
%ind_model_fit_exp
%modified T. Liefeld 06/12/92
%to fit an exponential model to data
%Further modified 6/96 MNeimark

j=j+1; % This is a run counter which indexes runs according
       % to their actual protocolled number.  Note that
       % i is an index only for non-calibrated runs.
rp=run_problem(j);

if (j==6|rp==1|rp==4|rp==6),
      j=j+1;  % Removes run number 6 which is a cal run or no data
end                  % runs.  NOTE:THIS CODE IS SPECIFIC FOR THE SLS-1 PROTOCOL.


%
% load data
%


eval(['load ',run_code,'.dec_spvl -mat']);
eval(['load ',run_code,'.dec_goodl -mat']);
eval(['load ',run_code,'.parms -mat']);

save_good = dec_good;
good_indices = find(dec_good==1);
if (spinv < 0)
      dec_spv = -dec_spv;
end

%
% Initialize time vector, assuming 4 Hz decimated frequency
%
l = length(dec_spv);
t = ([1:l] - 0.5) / 4;
t = t';

%
% shape tach signal with exponential (0.17 sec time constant)
% ramp to a steady state level at 'spinv'
%
Tv = 0.17;
if (rem(l,2) == 1)
   u = [ones(1,((l+1)/2)) zeros(1,((l-1)/2))];
else
   u = [ones(1,l/2) zeros(1,l/2)];
end
u = u';

%
% overall control input (tach)
%
u = lsim( spinv/Tv, [1, 1/Tv], u, t);

%
% Nominal model parameters.  The parameters to be fitted are the
% non-dimensional ratios of the physical parameters to the
% nominal model parameters here.  This places equal emphasis
% on each model parameter, even though they may be orders of
% magnitude apart.

K = .6;  % gain constant
T = 15;   %  time constant
A = -120;   % alpha_m  amplitude of step input   -- fixed
```

```
norm_parms = [K ; T ;  A];

options = [0 ; 0.001 ; 0.001];    %error tolerances -- see "help foptions"
vlb = [.1; .01; 1];  %lower bounds
vub = [10; 10; 1]; %upper bounds

plot(t(good_indices),dec_spv(good_indices))

%
% Fit the per-rotatory portion first
%

fprintf(['\n\n\nFitting ',run_code,' per-rotatory\n']);
dec_good = save_good;
dec_good(1:12) = zeros(1,12);         % do not fit first 3 seconds of data
dec_good(241:480) = zeros(1,240);     % do not fit post-rotatory data

if (sum(dec_good) < 10)
       fprintf('Not enough data points to determine a curve fit.\n');
       gain_vect(1,j)=NaN;
       tc_vect(1,j)=NaN;
       return;
end

good_indices = pack_true(dec_good);

model_parms = [1; 1; 1];
[model_parms, options] = constr('model_err_exp', model_parms, options, vlb, vub, [],
t, u, dec_spv, good_indices, norm_parms);

model_parms = model_parms .* norm_parms;
%eval(['save ',run_code,'.eperfit model_parms options'])

fprintf('*** Model fit: initial model parameters = 1.0\n');
fprintf('Number of iterations = %5.0f\n',options(10));
fprintf('Mean square error = %7.4f\n',options(8));

fprintf('K = %f\n',model_parms(1));
fprintf('T = %f\n',model_parms(2));
fprintf('A = %f\n',model_parms(3));
gain_vect(1,j)=model_parms(1);
tc_vect(1,j)=model_parms(2);
MSE_vect(1,j)=options(8)/model_parms(1);

%
% Fit the post-rotatory portion now
%

fprintf(['\n\n\nFitting ',run_code,' post-rotatory\n']);

dec_good = save_good(241:480);
dec_good(1:12) = zeros(1,12); % do not fit first 3 seconds of data

if (sum(dec_good) < 10)
       fprintf('Not enough data points to determine a curve fit.\n');
       gain_vect(2,j)=NaN;
       tc_vect(2,j)=NaN;
       return;
end
t=t(1:240);
dec_spv_p = dec_spv(241:480);
good_indices = pack_true(dec_good);
norm_parms = [K ; T ; -1*A];
model_parms = [1; 1; 1];
```

```
[model_parms, options] = constr('model_err_exp', model_parms, options, vlb, vub, [],
t, u, dec_spv_p, good_indices,norm_parms);

model_parms = model_parms .* norm_parms;
%eval(['save ',run_code,'.epostfit model_parms options'])

fprintf('*** Second fit: initial model parameters = 1.0\n');
fprintf('Number of iterations = %5.0f\n',options(10));
fprintf('Mean square error = %7.4f\n',options(8));

fprintf('K = %f\n',model_parms(1));
fprintf('T = %f\n',model_parms(2));
fprintf('A = %f\n',model_parms(3));
gain_vect(2,j)=model_parms(1);
tc_vect(2,j)=model_parms(2);
MSE_vect(2,j)=options(8)/model_parms(1);
```

```
% multiple_AATM
%
% written by T. Liefeld,  21/4/93
% modified by M. Neimark 6/96
% prepares a batch file for AATM processing using the
% appropriate calibration factors and names,  and a
% predefined batch file name, batch_factors, for use
% with OSfilt_diff_aatm.

for i = 1:number
      rp=run_problem(i);
      if ((i < 10) & (run(i) ~= 0 & (rp~=1 & rp~=4 & rp~=6))),
            n = num2str(i);
            run_code = [data_path,sub_code,'0',n];
            OSfilt_diff_aatm(run_code,hor_cal(i));
      else
            if (i >=10 & (run(i) ~= 0 & (rp~=1 & rp~=4 & rp~=6))),
                  n = num2str(i);
                  run_code = [data_path,sub_code,n];
                  OSfilt_diff_aatm(run_code,hor_cal(i));
            end
      end
end
```

```
function []=OSfilt_diff_aatm(run_code,cal)
% []=OSfilt_diff_attm(run_code,cal)
%
% This function calls upon OS filtering mex files to process raw position
% data.  Two PFMH's are performed, the filtered position data is then
% differentiated to obtain an eye velocity signal, and this is AATM filtered
% to remove saccades.  NOTE: THIS FUNCTION ASSUMES A 120 HZ SAMPLING RATE
% (used for most of the ground runs in rotating chair experiments)
% IF DATA IS NOT SAMPLED AT 120 HZ, USE THE RESAMPLE FUNCTION.
%MNeimark 6/96
% run_code=subject and run code used for file purposes
% cal=calibration factor

sample_rate=120;
dfilter=[0.0332, 0.0715, 0.0678, 0.0522, 0, -0.0522, -0.0678, -0.0715, ...
               -0.0332]; % a nine point differentiating filter

eval(['load ',run_code,'.eogh -mat']);
pos=OSPFMH(OSPFMH(eogh));  %PFMH filtering
clear eogh %free some memory
pos=pos*sample_rate*cal;
vel=fftfilt(dfilter,pos); % differentiate
clear pos
aspv=aatm(sample_rate,vel);
plot(aspv)
eval(['save ',run_code,'.aspv1 aspv -mat']);
```

```
/*PFMH.c*/
/*Code For mex interface script, pfmh.mex*/
#include <stdio.h>
#include <math.h>
#include "mex.h"

#ifndef DOUBLE
#ifdef THINK_C      /* THINK C doubles are extended, ints are short */
#define DOUBLE             double /* need true double for MATLAB matrices */
#define INT                int                    /* need long for user_fcn() */
#else               /* everyone else does the normal thing */
#define DOUBLE             double
#define INT                int
#endif
#endif

/* Input Arguments */

#define      POS_IN        prhs[0]


/* Output Arguments */

#define      POS_OUT             plhs[0]


#define      max(A,B)      ((A) > (B) ? (A) : (B))


#define      min(A,B)      ((A) < (B) ? (A) : (B))
#define N1 6
#define N2 10

void mexFunction(    /* Matlab mex-file interface routine */
       int nlhs,
       Matrix *plhs[],
       int nrhs,
       Matrix *prhs[]
       )
{
       void PFMH(), extend_pos();
       double *posi, *poso, *pos_temp;
       int m, n, num_samples;

       /* Check for proper number of arguments */
       if (nrhs != 1) {
              mexErrMsgTxt("PFMH requires one input argument.");
              }
       else if (nlhs != 1) {
              mexErrMsgTxt("PFMH requires one output argument.");
              }

       /* Ensure that VEL is a vector. */

       m = mxGetM(POS_IN);
       n = mxGetN(POS_IN);
       if ((max(m,n) <= 1) || (min(m,n) != 1)) {
              mexErrMsgTxt("PFMH requires that VEL be a vector.");
              }

       /* Create a matrix for the return argument */
       POS_OUT = mxCreateFull(m, n, REAL);

       /* Assign pointers to the various parameters */
```

```
        poso = mxGetPr(POS_OUT);
        posi = mxGetPr(POS_IN);

        /*Set up pos_temp, a version of the input position vector which is
appropriate*/
        /*For signal processing purposes*/
        num_samples = max(m,n);
        pos_temp=(double *)mxCalloc(num_samples+2*N2,sizeof(double));

        /* Do the actual computations in a subroutine */
        extend_pos(num_samples,posi,pos_temp);
        PFMH(num_samples, pos_temp, poso );
        /*Clear some Memory*/
        mxFree(pos_temp);
}

void extend_pos(
        int num_samples,
        double posi[],
        double pos_temp[]
        )
/* This function extends the matrix posi[] such that the function*/
/*PFMH will properly access indices (pos_temp is zero padded so that it is*/
/*more appropriate than posi for signal processing purposes).*/
{
        int i;

        for (i=0; i<N2; i++)
                pos_temp[i]=0.0;
        for (i=0; i<num_samples; i++)
                pos_temp[i+N2]=posi[i];
        for (i=0; i<N2; i++)
                pos_temp[i+N2+num_samples]=0.0;
}

void PFMH(
        int num_samples,
        double posi[],
        double poso[]
        )
{
        void PFMH1();
        double median5();
        int start, stop, i, j;
        double h1[N1], h2[N2];
        double x1F, x1B, x2F, x2B;

        PFMH1(N1,h1);
        PFMH1(N2,h2);
        start=N2;
        stop=num_samples+N2;
        for (i=start; i<stop; i++)
        {
                x1F=0;
                x1B=0;
                x2F=0;
                x2B=0;
                for (j=0; j<N1; j++)
                {
                        x1F += h1[j]*posi[i+1+j];
                        x1B += h1[j]*posi[i-1-j];
                }
                for (j=0; j<N2; j++)
                {
```

```
                        x2F += h2[j]*posi[i+1+j];
                        x2B += h2[j]*posi[i-1-j];
                }
                poso[i-start]=median5(posi[i],x1F,x1B,x2F,x2B);
        }
}

void PFMH1(
        int N,
        double h[]
        )
{
        int i;

        for (i=1; i<=N; i++)
                h[i-1]=(4*N-6*i+2)/(double)(N*(N-1));
}


double median5(
        double x1,
        double x2,
        double x3,
        double x4,
        double x5)
{
        double m;

        if (x1 >= x2) {         /* 1,2 */
                if (x2 >= x3) {         /* 1,2,3 */
                        if (x3 >= x4) {         /* 1,2,3,4 */
                                if (x3 >= x5) {
                                        m = x3;         /* 1,2,3,4,5 or 1,2,3,5,4 */
                                        }
                                else if (x2 >= x5) {
                                        m = x5;         /* 1,2,5,3,4 */
                                        }
                                else {
                                        m = x2;         /* 1,5,2,3,4 or 5,1,2,3,4 */
                                        }
                                }
                        else if (x2 >= x4) {/* 1,2,4,3 */
                                if (x4 >= x5) {
                                        m = x4;         /* 1,2,4,5,3 or 1,2,4,3,5 */
                                        }
                                else if (x2 >= x5) {
                                        m = x5;         /* 1,2,5,4,3 */
                                        }
                                else {
                                        m = x2;         /* 1,5,2,4,3 or 5,1,2,4,3 */
                                        }
                                }
                        else if (x1 >= x4) {/* 1,4,2,3 */
                                if (x2 >= x5) {
                                        m = x2;         /* 1,4,2,3,5 or 1,4,2,5,3 */
                                        }
                                else if (x4 >= x5) {
                                        m = x5;         /* 1,4,5,2,3 */
                                        }
                                else {
                                        m = x4;         /* 1,5,4,2,3 or 5,1,4,2,3 */
                                        }
                                }
                        else {                  /* 4,1,2,3 */
```

```
                if (x2 >= x5) {
                        m = x2;         /* 4,1,2,3,5 or 4,1,2,5,3 */
                        }
                else if (x1 >= x5) {
                        m = x5;         /* 4,1,5,2,3 */
                        }
                else {
                        m = x1;         /* 4,5,1,2,3 or 5,4,1,2,3 */
                        }
                }
        }
else if (x1 >= x3) {/* 1,3,2 */
        if (x2 >= x4) {      /* 1,3,2,4 */
                if (x2 >= x5) {
                        m = x2;         /* 1,3,2,4,5 or 1,3,2,5,4 */
                        }
                else if (x3 >= x5) {
                        m = x5;         /* 1,3,5,2,4 */
                        }
                else {
                        m = x3;         /* 1,5,3,2,4 or 5,1,3,2,4 */
                        }
                }
        else if (x3 >= x4) {/* 1,3,4,2 */
                if (x4 >= x5) {
                        m = x4;         /* 1,3,4,5,2 or 1,3,4,2,5 */
                        }
                else if (x3 >= x5) {
                        m = x5;         /* 1,3,5,4,2 */
                        }
                else {
                        m = x3;         /* 1,5,3,4,2 or 5,1,3,4,2 */
                        }
                }
        else if (x1 >= x4) {/* 1,4,3,2 */
                if (x3 >= x5) {
                        m = x3;         /* 1,4,3,2,5 or 1,4,3,5,2 */
                        }
                else if (x4 >= x5) {
                        m = x5;         /* 1,4,5,3,2 */
                        }
                else {
                        m = x4;         /* 1,5,4,3,2 or 5,1,4,3,2 */
                        }
                }
        else {                 /* 4,1,3,2 */
                if (x3 >= x5) {
                        m = x3;         /* 4,1,3,2,5 or 4,1,3,5,2 */
                        }
                else if (x1 >= x5) {
                        m = x5;         /* 4,1,5,3,2 */
                        }
                else {
                        m = x1;         /* 4,5,1,3,2 or 5,4,1,3,2 */
                        }
                }
        }
else {                  /* 3,1,2 */
        if (x2 >= x4) {      /* 3,1,2,4 */
                if (x2 >= x5) {
                        m = x2;         /* 3,1,2,4,5 or 3,1,2,5,4 */
                        }
                else if (x1 >= x5) {
                        m = x5;         /* 3,1,5,2,4 */
```

```
                                }
                        else {
                                m = x1;         /* 3,5,1,2,4 or 5,3,1,2,4 */
                                }
                        }
                else if (x1 >= x4) {/* 3,1,4,2 */
                        if (x4 >= x5) {
                                m = x4;         /* 3,1,4,5,2 or 3,1,4,2,5 */
                                }
                        else if (x1 >= x5) {
                                m = x5;         /* 3,1,5,4,2 */
                                }
                        else {
                                m = x1;         /* 3,5,1,4,2 or 5,3,1,4,2 */
                                }
                        }
                else if (x3 >= x4) {/* 3,4,1,2 */
                        if (x1 >= x5) {
                                m = x1;         /* 3,4,1,2,5 or 3,4,1,5,2 */
                                }
                        else if (x4 >= x5) {
                                m = x5;         /* 3,4,5,1,2 */
                                }
                        else {
                                m = x4;         /* 3,5,4,1,2 or 5,3,4,1,2 */
                                }
                        }
                else {                  /* 4,3,1,2 */
                        if (x1 >= x5) {
                                m = x1;         /* 4,3,1,2,5 or 4,3,1,5,2 */
                                }
                        else if (x3 >= x5) {
                                m = x5;         /* 4,3,5,1,2 */
                                }
                        else {
                                m = x3;         /* 4,5,3,1,2 or 5,4,3,1,2 */
                                }
                        }
                }
        }
else {                  /* 2,1 */
        if (x1 >= x3) {         /* 2,1,3 */
                if (x3 >= x4) {         /* 2,1,3,4 */
                        if (x3 >= x5) {
                                m = x3;         /* 2,1,3,4,5 or 2,1,3,5,4 */
                                }
                        else if (x1 >= x5) {
                                m = x5;         /* 2,1,5,3,4 */
                                }
                        else {
                                m = x1;         /* 2,5,1,3,4 or 5,2,1,3,4 */
                                }
                        }
                else if (x1 >= x4) {/* 2,1,4,3 */
                        if (x4 >= x5) {
                                m = x4;         /* 2,1,4,5,3 or 2,1,4,3,5 */
                                }
                        else if (x1 >= x5) {
                                m = x5;         /* 2,1,5,4,3 */
                                }
                        else {
                                m = x1;         /* 2,5,1,4,3 or 5,2,1,4,3 */
                                }
                        }
```

```
        else if (x2 >= x4) {/* 2,4,1,3 */
                if (x1 >= x5) {
                        m = x1;       /* 2,4,1,3,5 or 2,4,1,5,3 */
                        }
                else if (x4 >= x5) {
                        m = x5;       /* 2,4,5,1,3 */
                        }
                else {
                        m = x4;       /* 2,5,4,1,3 or 5,2,4,1,3 */
                        }
                }
        else {              /* 4,2,1,3 */
                if (x1 >= x5) {
                        m = x1;       /* 4,2,1,3,5 or 4,2,1,5,3 */
                        }
                else if (x2 >= x5) {
                        m = x5;       /* 4,2,5,1,3 */
                        }
                else {
                        m = x2;       /* 4,5,2,1,3 or 5,4,2,1,3 */
                        }
                }
        }
else if (x2 >= x3) {/* 2,3,1 */
        if (x1 >= x4) {     /* 2,3,1,4 */
                if (x1 >= x5) {
                        m = x1;       /* 2,3,1,4,5 or 2,3,1,5,4 */
                        }
                else if (x3 >= x5) {
                        m = x5;       /* 2,3,5,1,4 */
                        }
                else {
                        m = x3;       /* 2,5,3,1,4 or 5,2,3,1,4 */
                        }
                }
        else if (x3 >= x4) {/* 2,3,4,1 */
                if (x4 >= x5) {
                        m = x4;       /* 2,3,4,5,1 or 2,3,4,1,5 */
                        }
                else if (x3 >= x5) {
                        m = x5;       /* 2,3,5,4,1 */
                        }
                else {
                        m = x3;       /* 2,5,3,4,1 or 5,2,3,4,1 */
                        }
                }
        else if (x2 >= x4) {/* 2,4,3,1 */
                if (x3 >= x5) {
                        m = x3;       /* 2,4,3,1,5 or 2,4,3,5,1 */
                        }
                else if (x4 >= x5) {
                        m = x5;       /* 2,4,5,3,1 */
                        }
                else {
                        m = x4;       /* 2,5,4,3,1 or 5,2,4,3,1 */
                        }
                }
        else {              /* 4,2,3,1 */
                if (x3 >= x5) {
                        m = x3;       /* 4,2,3,1,5 or 4,2,3,5,1 */
                        }
                else if (x2 >= x5) {
                        m = x5;       /* 4,2,5,3,1 */
                        }
```

```
                                else {
                                        m = x2;         /* 4,5,2,3,1 or 5,4,2,3,1 */
                                        }
                                }
                        }
                else {                  /* 3,2,1 */
                        if (x1 >= x4) {         /* 3,2,1,4 */
                                if (x1 >= x5) {
                                        m = x1;         /* 3,2,1,4,5 or 3,2,1,5,4 */
                                        }
                                else if (x2 >= x5) {
                                        m = x5;         /* 3,2,5,1,4 */
                                        }
                                else {
                                        m = x2;         /* 3,5,2,1,4 or 5,3,2,1,4 */
                                        }
                                }
                        else if (x2 >= x4) {/* 3,2,4,1 */
                                if (x4 >= x5) {
                                        m = x4;         /* 3,2,4,5,1 or 3,2,4,1,5 */
                                        }
                                else if (x2 >= x5) {
                                        m = x5;         /* 3,2,5,4,1 */
                                        }
                                else {
                                        m = x2;         /* 3,5,2,4,1 or 5,3,2,4,1 */
                                        }
                                }
                        else if (x3 >= x4) {/* 3,4,2,1 */
                                if (x2 >= x5) {
                                        m = x2;         /* 3,4,2,1,5 or 3,4,2,5,1 */
                                        }
                                else if (x4 >= x5) {
                                        m = x5;         /* 3,4,5,2,1 */
                                        }
                                else {
                                        m = x4;         /* 3,5,4,2,1 or 5,3,4,2,1 */
                                        }
                                }
                        else {                  /* 4,3,2,1 */
                                if (x2 >= x5) {
                                        m = x2;         /* 4,3,2,1,5 or 4,3,2,5,1 */
                                        }
                                else if (x3 >= x5) {
                                        m = x5;         /* 4,3,5,2,1 */
                                        }
                                else {
                                        m = x3;         /* 4,5,3,2,1 or 5,4,3,2,1 */
                                        }
                                }
                        }
                }
        return(m);
}
```

```
% Script Run_Report_Maker:
% MNeimark 9/96
% For a session, this script generates run reports for generating
% hard copy records of the testing in a session.

num_runs=length(run);
run_num_vect=1:num_runs;
session=str2num(sub_code(2));
if length(sub_code)==3,
       session=str2num(sub_code(2:3));
end

for i=1:num_runs,
       rp=run_problem(i);
       tcper=tc_vect(1,i);
       tcpost=tc_vect(2,i);
       gainper=gain_vect(1,i);
       gainpost=gain_vect(2,i);
       pgper=pgood_vect(1,i);
       pgpost=pgood_vect(2,i);
       msper=MSE_vect(1,i);
       mspost=MSE_vect(2,i);

       if (run(i) == 1) & (rp~=1) & (rp~=4) & (rp~=6),
             fprintf(fid,'Subject:                            %c\n',sub_code(1));
             fprintf(fid,'Session:                            %d\n',session);
             fprintf(fid,'Run:                                %d\n',i);
             if session<=5
                    fprintf(fid,'Condition:                          PRE\n');
             elseif (session==6|session==7|session==8)
                    fprintf(fid,'Condition:                          EARLY\n');
             elseif (session==9 | session==10)
                    fprintf(fid,'Condition:                          LATE\n');
             else
                    fprintf(fid,'Condition:                          OTHER\n');
             end
             if (i==2|i==4|i==7|i==9),
                    if run_problem(i)~=7
                           fprintf(fid,'Direction:                         CW\n');
                    else
                           fprintf(fid,'Direction:                         CC\n');
                    end
             elseif (i==3| i==5| i==8| i==10),
                    if run_problem(i)~=7,
                           fprintf(fid,'Direction:                         CC\n');
                    else
                           fprintf(fid,'Direction:                         CW\n');
                    end
             else
                    fprintf(fid,'Direction:                              OTHER\n');
             end
             if rp~=5,
                    if (i==4|i==5|i==9|i==10),
                           fprintf(fid,'Head Position:                     DMP\n');
                    else
                           fprintf(fid,'Head Position:                     HE\n');
                    end
             else
                    fprintf(fid,'Head Position:                      HE\n');
             end
             fprintf(fid,'\n\n');

             if (i < 10)
                    n = [num2str(0),num2str(i)];
```

```
          else
                n = num2str(i);
          end
          code= [data_path,sub_code,n];
          eval(['load ',code,'.parms -mat']);

          fprintf(fid,'Spin Length:                            %4.2f\n',spinl);
          fprintf(fid,'Run Length:                             %3.0f\n',runlen);
          fprintf(fid,'Spin Velocity:                          %4.1f\n',spinv);
          fprintf(fid,'\n\n');
          fprintf(fid,'LL Per-Rot Gain:                        %4.3f\n',gainl);
          fprintf(fid,'LL Per-Rot TC:                          %4.2f\n',taul);
          fprintf(fid,'MOD Per-Rot Gain:                       %4.3f\n',gainper);
          fprintf(fid,'MOD Post-Rot TC:                        %4.2f\n',tcper);
          fprintf(fid,'Per-Rot %%Good:                          %d\n',pgper);
          fprintf(fid,'Per-Rot MSE:                            %4.2f\n',msper);
          fprintf(fid,'\n\n');
          fprintf(fid,'LL Post-Rot Gain:                       %4.3f\n',gain2);
          fprintf(fid,'LL Post-Rot TC:                         %4.2f\n',tau2);
          fprintf(fid,'MOD Post-Rot Gain:                      %4.3f\n',gainpost);
          fprintf(fid,'MOD Post-Rot TC:                        %4.2f\n',tcpost);
          fprintf(fid,'Post-Rot %%Good:                         %d\n',pgpost);
          fprintf(fid,'Post-Rot MSE:                           %4.2f\n',mspost);
          fprintf(fid,'\n\n\n\n');
          clg
          eval(['load ',code,'.dec_spv1 -mat']);
          eval(['load ',code,'.dec_good1 -mat']);
          axis([0 120 -150 150]);
          sample = 4;
          minute_size = 60 * sample;
          t = [1:(2*minute_size)+1] / sample;
          xlabel('Time since chair start (sec)');
          ylabel('Slow Phase Velocity (deg/sec)');
          plot(t(dec_good),dec_spv(dec_good),'.');
          hold on

          T(1)=tcper; K(1)=gainper;
          T(2)=tcpost; K(2)=gainpost;
          A(1)=120; A(2)=-120;
          if spinv>0
                A=-A;
          end
          x=A(1)*K(1)*exp(-1*t/T(1));
          plot(t(1:240),x(1:240));

          y=A(2)*K(2)*exp(-1*t/T(2));
          plot(t(241:480),y(1:240));

          hold off
          title(['SLS1 ',sub_code,n,' Fit']);
          eval(['print ',sub_code,n,'.ps']);
     end
end
```

```
% Script sav_dat_to_file
%
% This script utilizes data from the model paramater fits, and labels representing
% subject, run, session, spin direction, spin duration, testing conditions
% (preflight, postflight, inflight) and inserts all (line by line for each segment)
% into a text data base which will be imported into Systat for analysis.
% MNeimark 6/96

num_runs=length(run);
run_num_vect=1:num_runs;
session=str2num(sub_code(2));
if length(sub_code)==3,
        session=str2num(sub_code(2:3));
end

for i=1:num_runs,
        rp=run_problem(i);
        if (run(i) == 1) & (rp~=1) & (rp~=4) & (rp~=6),
                % Test to ensure that run is to be recorded
                pgoodper=pgood_vect(1,i);
                pgoodpost=pgood_vect(2,i);

                tcper=fix(tc_vect(1,i)*100)/100;
                tcpost=fix(tc_vect(2,i)*100)/100;
                gainper=fix(gain_vect(1,i)*100)/100; %For Constraint Outliers
                gainpost=fix(gain_vect(2,i)*100)/100;
                MSEper=MSE_vect(1,i);
                MSEpost=MSE_vect(2,i);

                % Work on the per-rot segment first
                if pgoodper>=60 & (tcper>0.15) & (gainper>0.06)...
                    & (tcper<150) & (gainper<6) & (MSEper<500),
                        fprintf(fid,'%c ',sub_code(1));
                        fprintf(fid,'PER ');
                        if session<=5
                                fprintf(fid,'PRE ');
                        elseif (session==6|session==7|session==8)
                                fprintf(fid,'EARLY ');
                        elseif (session==9 | session==10)
                                fprintf(fid,'LATE ');
                        else
                                fprintf(fid,'OTHER ');
                        end
                        if (i==2|i==4|i==7|i==9),
                                if run_problem(i)~=7
                                        fprintf(fid,'CW ');
                                else
                                        fprintf(fid,'CC ');
                                end
                        elseif (i==3| i==5| i==8| i==10);
                                if run_problem(i)~=7
                                        fprintf(fid,'CC ');
                                else
                                        fprintf(fid,'CW ');
                                end
                        else
                                fprintf(fid,'OTHER ');
                        end
                        if (~isnan(tcper) & run_problem(i)~=2),
                                fprintf(fid,'%4.2f ',tcper);
                        else
                                fprintf(fid,'- ');
                        end
                        if ~isnan(gainper) & run_problem(i)~=2,
```

```
              fprintf(fid,'%4.4f ',gain_vect(1,i));
       else
              fprintf(fid,'- ');
       end
       fprintf(fid,'%d ',i);
       fprintf(fid,'%d ',session);
       if (sub_code(1)=='T' |sub_code(1)=='M')
              fprintf(fid,'Y\n');
       else
              fprintf(fid,'N\n');
       end
end
% Now work on post rot segment
if pgoodpost>=60 & (tcpost>0.15) & (gainpost>0.06)...
   & (tcpost<150) & (gainpost<6) & (MSEpost<500),
       fprintf(fid,'%c ',sub_code(1));
       if rp~=5,
              if (i==4|i==5|i==9|i==10),
                     fprintf(fid,'DMP ');
              else
                     fprintf(fid,'HE ');
              end
       else
              fprintf(fid,'HE\n');
       end
       if session<=5
              fprintf(fid,'PRE ');
       elseif (session==6|session==7|session==8)
              fprintf(fid,'EARLY ');
       elseif (session==9 | session==10)
              fprintf(fid,'LATE ');
       else
              fprintf(fid,'OTHER ');
       end
       if (i==2|i==4|i==7|i==9),
              if run_problem(i)~=7
                     fprintf(fid,'CW ');
              else
                     fprintf(fid,'CC ');
              end
       elseif (i==3| i==5| i==8| i==10),
              if run_problem(i)~=7,
                     fprintf(fid,'CC ');
              else
                     fprintf(fid,'CW ');
              end
       else
              fprintf(fid,'OTHER ');
       end
       if ~isnan(tcpost) & run_problem(i)~=3,
              fprintf(fid,'%4.2f ',tcpost);
       else
              fprintf(fid,'- ');
       end
       if ~isnan(gainpost) & run_problem(i)~=3
              fprintf(fid,'%4.4f ',gain_vect(2,i));
       else
              fprintf(fid,'- ');
       end
       fprintf(fid,'%d ',i);
       fprintf(fid,'%d ',session);
       if (sub_code(1)=='T' |sub_code(1)=='M')
              fprintf(fid,'Y\n');
       else
```

```
                                        fprintf(fid,'N\n');
                                end
                        end
                end
        end
```

```
%stat_prep_batch
%
% Prepares an SPV profile for statistical analysis.  The first
% step is time-shifting and stripping out extra data to leave
% one minute per-rotatory and one minute post-rotatory.  The
% second step is outlier detection.  The third step is decimation
% by a factor of 30 down to 4 Hz.
%
% D. Balkwill 8/8/91
%
% Modified by T Liefeld, 12/17/92    to use a longer per-
% rotatory period consistent with the later use of actual
% tach signals as the stimulus for per-rotatory model
% fitting scripts.
% Modified again for use with batch analysis scripts
% Further Modified 6/96 MNeimark

sample = 120;
minute_size = 60 * sample;

%load data
eval(['load ',run_code,'.aspvl -mat']);
eval(['load ',run_code,'.parms -mat']);

x = aspv;
clear aspv

%
% Normalize SPV profile to one-minute per and post-rotatory
% On the longer per_rotatory file,  perform outlier detection
% only upon the normslized one minute section due to the
% difficulty in dealing with a different exponential on the
% initial rise.  Make all extended per-rot files 65 seconds
% long, truncating extra data or padding as before

y = zeros(1,2 * minute_size + 1);   %initialize to two minutes

delay = delay*sample ;
spinl = spinl*sample;

if (spinl >= minute_size)  %extract first minute of per-rotatory
    y(1:minute_size) = x(delay:(delay+minute_size-1));
else     %pad per-rotatory out to one minute
    y(1:spinl) = x(delay:(delay+spinl-1));
    y(spinl+1:minute_size) = zeros(1,minute_size-spinl);
end


%post-rotatory data

if ((max(size(x))-delay-spinl) >= minute_size)  %extract first minute
                                                % of post-rotatory
    y(minute_size+1:2*minute_size+1) = x((delay+spinl):(delay+spinl+minute_size));


else     %pad post-rotatory out to one minute

   y((minute_size+1):(minute_size+max(size(x))-delay-spinl)) =
x((delay+spinl+1):(max(size(x))));
    y((minute_size+1+max(size(x))-delay-round(spinl)):2*minute_size+1) =
zeros(1,minute_size-max(size(x))+1+delay+round(spinl)) * median(x((max(size(x))-
5):(max(size(x))-1)));
end
```

```
%
% Determine sections to be excluded from statistical analysis
% (dropouts and outliers).
%

t = [1:(2*minute_size+1)] / sample;
good_data = ones(1,2*minute_size+1);

%
%find valid range for log outlier detection in per-rotatory section
%
fprintf('Per-rotatory:\n');
i = sample + 1;  %one second after start
j = i + 20 * sample;    %insist upon 20 seconds, minimum
while (abs(mean(y(j:j+5*sample))) > 10)  %look for mean spv under 10 deg/sec
    j = j + 2 * sample;
end
t1 = t(i:j);
y1 = y(i:j);

[logout1,under,over,m,b] = log_outlier(t1,y1);

%take care of under- or over-flow
good_data(i:j) = ~logout1;
if (under > 0)
    good_data(i-under:i-1) = zeros(1,under);
end
if (over > 0)
    good_data(j+1:j+over) = zeros(1,over);
end

%save final fit in parms file as first-order "model fit"
tau1 = -1/m;
gain1 = exp(b)/120;
fprintf('Time length of outliers from log fit is ');
fprintf('%5.2f seconds.\n',(sum(logout1)+under+over)/sample);

% do magnitude outlier detection on remainder of per-rotatory SPV

i = j + over + 1;
[magout1,under,over] = mag_outlier(t(i:minute_size),y(i:minute_size),30);
good_data(i:minute_size) = ~magout1;
if (under > 0)
    good_data(i-under:i-1) = zeros(1,under);
end
fprintf('Time length of outliers from magnitude threshold is');
fprintf(' %5.2f seconds.\n',(sum(magout1)+under)/sample);
% don't fill in any overflow, because this would be post-rotatory

delay = delay / sample;
spin1 = spin1 / sample;

%
%find valid range for outlier detection in post-rotatory section
%
fprintf('Post-rotatory:\n');
i = minute_size + sample + 1;  %one second after stop
j = i + 20 * sample;    %insist upon 20 seconds, minimum
while (abs(mean(y(j:j+5*sample))) > 10)
    j = j + 2 * sample;
      if j>2*minute_size+1-5*sample,  % Added 8/16 1996 MAN
             j=2*minute_size+1-5*sample; %  to avoid error.
             break
      end
```

```
end
t2 = t(i:j);
y2 = y(i:j);

[logout2,under,over,m,b] = log_outlier(t2,y2);

%take care of under- or over-flow
good_data(i:j) = ~logout2;
if (under > 0)
    good_data(i-under:i-1) = zeros(1,under);
end
if (over > 0)
    good_data(j+1:j+over) = zeros(1,over);
end

%save final fit in parms file as first-order "model fit"
tau2 = -1/m;
gain2 = exp(b + 60 * m)/120;
fprintf('Time length of outliers from log fit is ');
fprintf('%5.2f seconds.\n',(sum(logout2)+under+over)/sample);

% do magnitude outlier detection on remainder of post-rotatory SPV

i = j + over + 1;
[magout2,under,over] = mag_outlier(t(i:2*minute_size),y(i:2*minute_size),30);
good_data(i:2*minute_size) = ~magout2;
if (under > 0)
    good_data(i-under:i-1) = zeros(1,under);
end
fprintf('Time length of outliers from magnitude threshold');
fprintf(' is %5.2f seconds.\n',(sum(magout2)+under)/sample);
% don't fill in any overflow, because this would be past two minutes

fprintf('Overall percentage of good data is %6.2f\n',100*mean(good_data));

eval(['save ',run_code,'.parms  delay spin1 spinv runlen tau1 gain1 tau2 gain2 T1
T2']);

clear t t1 t2 y1 y2 i j logout1 logout2 minute_size sample
clear delay spin1 spinv T1 T2 runlen under over magout1 magout2
clear m b tau1 gain1 tau2 gain2

norm_spv = y;

clear y

% save normalized data, having departed from 'file_specs' at this point

dec_spv=resample(norm_spv,4,120);
dec_good=round(resample(good_data,4,120));

eval(['save ',run_code,'.dec_spv1 dec_spv']);
eval(['save ',run_code,'.dec_good1 dec_good']);

clear good_data norm_spv
clear t tau1 tau2 run_len spin1 spinv
```

DIRECTORY
OF CSR
EMPLOYEES

1996-1997

Massachusetts Institute of Technology
77 Massachusetts Avenue
Cambridge, MA 02139

CSR

Center for Space Research

NAME	EXTENSION	OFFICE	USERID
ALAN, Douglas	3-3987	37-555	nessus
ASSAD, Albert	3-0017	37-307	albert
ATHENS, Demitrios	8-5285	NE80-6059	da
AUCOIN, Richard	3-3942	37-495	rja
AVRUCH, Ian	3-2551	26-348	avruch@maggie
AMIR, Amir R.	3-0017	37-307	aramir@mit
BAGANOFF, Frederick	3-6892	37-518A	fkb
BALKWILL, Barbara A.	3-6104	37-287	balkwill
BALUTA, Christopher	8-7348	37-662C	baluta
BARILARO, Kristen	3-6116	37-276	kfisher
BAUTZ, Mark	3-0023	37-521	mwb
BEALL, Andrew	3-7758	37-151	beall
BECKER, Christopher	3-7554	37-432	cmbecker
BECKER-HAARSMA, Deborah	3-2571	26-345	djbecker@maggie
BELCHER, John W. (Prof.)	3-4285	37-695	jwb
BENFORD, Richard L.	3-8062	37-275	benford
BERTSCHINGER, Edmund (Prof.)	3-5083	6-207	edbert@arcturus
BLAKESLEE, John	3-6094	6-218M	john@alioth
BLOZIE, Robert	3-8657	NE80-6019	blozie
BOCHNER, Brett	3-0203	20B-145	brett@tristan
BOUGHAN, Edward A.	3-6146	NE80-6069	eab
BOUGHAN, Joan G.	3-6102	37-287	jgb
BRADT, Hale V.D. (Prof.)	3-7550	37-587	hale
BRADY, Tye	3-5089	37-501	tye
BRAVER, Alan	3-7960	NE80-6009	alan
BRESLAU, David	3-7216	37-664	breslau
BRIGGS, Mary	8-7896	NE80-6048	briggs
BROWN, Sharon Leah	3-8809	37-355	slbrown@mit
BUEHLER, Royce	3-9766	37-581	buehler
BURGESS, Ralph	3-8121	20B-145	none
BURKE, Bernard F. (Prof.)	3-2572	26-335	bfburke@athena
CALLAGHAN, Sean	3-2890	NE80-6048	cal
CANIZARES, Claude R. (Prof.)	3-7501	37-241	crc
CARTWRIGHT, John	3-3074	26-349	skc@athena
CHAKRABARTY, Deepto	3-3840	37-561	deepto
CHANG, Tom T.S.	3-7523	37-261	tsc
CHATTERJI, Shourov	3-0203	20F-107	shourov@tristan
CHEN, Grace	3-3074	26-344	gchen@athena
CLARK, George W. (Prof. Emeritus)	3-5842	37-611	gwc
CONKLIN, Anne	3-5628	26-331	aconklin@mit
COUNSELMAN, Charles C. (Prof.)	3-7902	37-552	ccc
CRAWFORD, Fronfield	3-3074	26-349	crawford@athena
CREW, Geoffrey B.	3-3789	37-515	gbc
CRISAFULLI, Janice	3-2890	NE80-6088	jmc
CSATORDAY, Peter	3-0203	20B-145	peter@tristan
CUI, Wei	3-6996	37-571	cui

NAME	EXTENSION	OFFICE	USERID
DAVIS, David	3-8758	37-662B	dsd
DAVIS, John	8-8119	37-662D	davis
DEWEY, Daniel	3-7244	37-635	dd
DOTY, John	3-3169	37-541	jpd
DOUCETTE, Michael	3-3571	NE80-6047	mcd
EDWARDS, Peggy	8-7390	37-367	margaret@mit
ELDER, Richard	8-7481	NE80-6079	relder
ELLITHORPE, John	3-3074	26-349	jdell@maggie
ENRIGHT, Michael	3-8009	NE80-6088	none
ERNSTMEYER, James	3-7373	37-294	imjay
EVANS, Thomas	3-4824	20B-145C	evans@tristan
FARRELL, Kimberly	3-1736	37-607	kimf
FERGASON, Beverly	8-8153	NE80-6005	fergason
FINCK, Leslie	3-2401	37-684B	laf
FLANAGAN, Kathryn	8-7324	NE80-6081	kaf
FLEMING, Robert	3-3130	37-411	rcf
FLETCHER, Andre	3-3074	26-344	abfletch@athena
FORBES, William	3-2801	NE80-6026	none
FORD, Peter G.	3-6485; 3-7277	37-601; NE80-6059	pgf
FOSTER, Richard	3-6808	NE80-6063	rickf
FOX, Derek	3-7457	37-618	derekfox
FRANCIS, James	3-2044	NE80-6029	jimf
FRITSCHEL, Peter	3-8153	20F-101C	pf@tristan
GAGE, Deborah	3-0228	NE80-6043	dgage
GAIDOS, Eric	3-7457	37-618B	ejgaidos
GALTON, Eugene G.	3-7294, 3-7502	37-415, 37-241	ebg
GOEKE, Robert F.	3-1910	NE80-6035	goeke
GONG, Gordon	3-8135	NE80-6055	ggg
GONZALEZ, Gabriela	3-6410	20F-104	gg@taliesin
GORDON, Dorothy	3-0797	NE80-6053	dag
GOVAERT, Johan	3-1977	NE80-6049	hansg
GRIFFITH, Mark	3-4028	26-319	mrg@athena
GRUPP, Deborah	3-0698	37-274	grupp
HANLON, Rosemary	3-8433	37-538	hanlon
HARPAZ, Amos	3-0905	6-211	harpaz@brmha
HASTREITER, Dawn	3-7509	37-155	hast@mit
HEROLD, Lori	3-3074	26-344	lherold@maggie
HERTZ, Arlyn	3-1456	37-241	aph
HEWITT, Jacqueline N. (Prof.)	3-3071	26-327	jhewitt@athena
HINDLE, Patrick	3-8764	37-420	psh
HOHLFELD, Kathleen	3-3322	37-422C	kelly
HUENEMOERDER, David	3-4283	37-667	dph
HUGHES, John	3-8009	NE80-6077	none

NAME	EXTENSION	OFFICE	USERID
ISOBE, Takashi	3-7396	37-558	ti
JACKSON, Keoki	3-5487	37-301	dkeokij@mit
JASPERSE, John	3-7373	37-294	**none**
JONES, Stephen	3-0320	37-662D	sjones
JONES, Tim	3-8740	37-410	mtj
JOSS, Paul C. (Prof.)	3-4845	6-203	joss@mitlns
KASPARIAN, Fred	3-1065	NE80-6099	kaspar
KASTNER, Joel	3-3875	37-667	jhk
KATZ, Charlie	3-3074	26-344	ckatz@athena
KINTNER, Eric	8-8391	37-227	ekintner@mit
KISSEL, Steve	3-7242	37-501	sek
KLATT, Brian	3-7555	NE80-6037	bk
KOMMERS, Jeffrey	3-7357	37-618	kommers
KRUZEL, Edward	8-6832	20F-002	elk@tristan
LALIBERTE, Robert	3-2705	NE80-6018	liberty
LANTZ, Brian	3-0203	20F-111	brian@tristan
LAZARUS, Alan J.	3-4284	37-687	ajl
LEVINE, Alan M.	3-7525	37-575	aml
LEWIN, Walter H.G. (Prof.)	3-4282	37-627	lewin
LINTON, Beverly	3-3769	37-675	balinton
LYONS, Mitchell	no phone	NE80-6105	lyons
MacINNIS, Myron	3-2705	NE80-6018	myron
McGUIRK, Michael	3-3722	37-491	mm
McMAHON, Pauline	3-8130	26-347	**none**
MACHACEK, Marie	3-0690	6-206	mariem@achernar
MANNING, Herbert	8-8296	37-558	hlm
MARIN, Alexandru	3-6413	20F-101A	marin@ligo
MAROLDA, James	3-7941	37-280	**none**
MARSHALL, Herman	3-8573	37-667	hermanm
MARTEL, Francois	3-9847	37-511	fm
MAVALVALA, Nergis	3-4824	20B-145A	nergis@tristan
MAYER, William F.	3-7552	NE80-6043	wfm
MILLEN, Roger	3-9716	37-632	millen
MILLER, David	3-3288	37-371	millerd@mit
MILLER, Frederick	3-3825	37-540	frd
MILLIGAN, Pamela	3-7217	37-686	pam
MONTGOMERY, John	8-8108	NE80-6061	johnm
MOORE, Christopher	3-4028	26-319	cmoore@alioth
MORGAN, Edward	3-3182	37-567	ehm
NATAPOFF, Alan	3-7757	37-147	natapoff
NEIMARK, Matt	8-5794	37-134	mattn@mit
NEWMAN, Dava, (Prof.)	8-8799	33-119	dnewman

NAME	EXTENSION	OFFICE	USERID
O'CONNOR, James R.	3-6139	NE80-6087	joc
OMAN, Charles M.	3-7508	37-211	cmo
PAK, Christopher	3-9342	37-487	crispy
PATEL, Virendra	no phone	NE80-6065	viren
PAULARENA, Karolen	3-5877	37-651	kip
PECHE, Paula	2-1536	37-315	ppeche@mit
PETTENGILL, Gordon (Prof. Emeritus)	3-4281	37-641	ghp
PHILLIPS, Kenton	3-2067	37-422C	kenton
PIVOVAROFF, Michael	3-7466	37-524	mjp
PLUMMER, William	3-4824	20B-145	will@tristan
POLITANO, Jr., John	3-6103	37-287	jpjr
PORTER, Jeanne	8-8615	37-484	porter
PRENTISS, Jane	8-8615	37-484	prentiss
PRIGOZHIN, Gregory	3-7246	37-662D	gyp
QUIGLEY, Joan	3-2490/3-4287	37-662A	jcq
RAPPAPORT, Saul A. (Prof.)	3-7551	37-551	sar
RASIO, Frederic (Prof.)	3-5084	6-201	rasio@mit
RASMUSSEN, Scott	494-2645	Volpe	scottr@mit
REMILLARD, Ronald	3-2786	37-595	rr
RENSHAW, Robert	8-8090	NE80-6013	bobr
RETTERER, John	3-7527	37-662B	jmr
RICHARD, Michael	3-4824; 3-3746	20B-145; 37-582	mjr@tristan
RICHARDSON, John D.	3-6112	37-655	jdr
RICKER, George R.	3-7532	37-535	grr
ROCHLIS, Jennifer	3-5487	37-301	jrochlis@mit
RONG, Haisheng	8-8279	20B-145	rong@tristan
SAHA, Partha	3-4824	20B-145C	partha@tristan
SANTIAGO, Teresa	3-7078	37-284	teresa
SAUST, Anne Berit	3-8130	26-347	saust@maggie
SCHAFFNER, Grant	3-0017	37-307	grant@umhlanga
SCHATTENBURG, Mark	3-3180	37-421	marks
SCHECHTER, Paul (Prof.)	3-0690	6-206	schech@achernar
SCHMIDT, Patricia	3-2628	37-311	schmidtp@athena
SCHULZ, Norbert	8-5767	37-644	nss
SEN, Ellen	3-8342	NE80-6103	esen
SHIREY, Robert	3-7554	37-432	shirey
SHOEMAKER, David	3-6411	20F-111A	dhs@tristan
SIGG, Daniel	8-8295	20F-104	sigg@tristan
SISSON, Robert	8-8615	37-484	sisson
SKWERSKY, Adam	800/439-2870(3-7505)	37-135	askwersk@mit
SLIWA, Krzysztof	3-4824	20B-145	***none***
SMITH, Matthew	2-1736	NE80-6003	matt
SMITH, Robin	3-2628	37-311	rlsmith@athena

NAME	EXTENSION	OFFICE	USERID
STEINBERG, John T.	3-2354	37-680	jts
TAM, Wing-Yee (Sunny)	3-7373	37-294	wyt
TAPPAN, John	3-3319	NE80-6095	jht
TERHUNE, Mary	3-7527	37-271	mt
TETREAULT, David	3-3849	37-662B	djt
TOMASSINI, Anna	3-7509	37-155	anita@athena
TOVEE, Christine	3-7509	37-155	catovee@mit
TRYFONIDIS, Michail	3-2628	37-311	miket@mit
VANDERSPEK, Roland	3-8456	37-527	roland
VEZIE, Michael	8-8954	37-426	dv
VILLASENOR, Joel	2-1667	37-414	jsvilla
VINING, Joanne	8-8090	NE80-6013	jsv
WANG, Chi	3-2401	37-684A	cw
WARREN, Edward	3-1366	NE80-6077	ewarren
WEISS, Rainer (Prof.)	3-3527	20F-102	weiss@tristan
WISE, Michael	8-7254	37-644	wise
WOJDOWSKI, Patrick	3-7457	37-624C	pswoj
WOOD, Alan	8-6861	37-576	chekov
WRIGHT, Ed	8-7896	NE80-6080	none
YASSEEN, Fareed	3-7373	37-294	fy
YOUNG, Laurence R. (Prof.)	3-7759	37-207	lry
ZUCKER, Michael	3-8070	20F-101B	zucker_m@ligo
CSR COMMON AREAS IN NE80:			
Pump Room	3-2705	NE80-6018	
Machine Shop	3-2801	NE80-6026	
Flight Assembly	3-8009	NE80-6088	
General Electronics	3-2890	NE80-6048	
Project Lab	3-2943	NE80-6078	
Mardix Booth	3-7020	NE80-Ground	
Loading Dock	182-8-6040	NE80-Ground	
NE80 FAX Number:	3-8084	NE80-6044	

SUPPORT STAFF WITHIN CSR
BUILDINGS 37, NE80, 20, 6, and 26

OCATION	NAME	ROOM	EXTEN.	USERID
UILDING 37				
2ND FLOOR	Secretary/MVL	37-219	3-7805	mvl
	Arlyn Hertz	37-241	3-1456	aph
	Mary Terhune	37-271	3-7527	mt
	Teresa Santiago	37-284	3-7078	teresa
	Kristen Barilaro	37-276	3-6116	kfisher
	Barbara Balkwill	37-287	3-6104	balkwill
5TH FLOOR	Rosemary Hanlon	37-538	3-8433	hanlon
	Michael Richard	37-582	3-3746	mjr
6TH FLOOR	Beverly Linton	37-675	3-3769	balinton
	Kimberly Farrell	37-607	3-1736	kimf
LDG. NE80	Deborah Gage	NE80-6044	3-0228	dgage
UILDING 6	Secretary	6-204	3-3718	
UILDING 20	Michael Richard	20B-145	3-4824	mjr
	Will Plummer	20B-145	3-4824	will
UILDING 26	Ann Conklin	26-331	3-5628	aconklin@mit

CSR ADMINISTRATION

	ROOM	EXTENSION
Director's Office		
Claude R. Canizares, Director	37-241	3-7501
Eugene B. Galton, Acting Associate Director	37-255	3-7502
Arlyn Hertz, Administrative Secretary	37-241	3-1456
FAX NUMBER: 253-3111		
Administrative Office		
John P. Politano, Jr., Administrative Officer	37-291	3-6103
Barbara A. Balkwill, Administrative Secretary	37-287	3-6104
Deborah Grupp, Financial Assistant	37-274	3-0698
Joan G. Boughan, Fiscal Officer	37-281	3-6102
Teresa Santiago, Fiscal Administrative Assistant	37-284	3-7078
Kristen Barilaro, Senior Secretary	37-276	3-6116
CSR PRIMARY FAX NUMBER: 253-0861		
CSR Facility Office		
Richard Benford, Safety & Facility Coordinator	37-275	3-8062
James Marolda, Driver/Utility	37-280	3-7941
Computer & Network Information		
Kenton Phillips, Network Manager	37-422C	3- 2067
Kathleen Hohlfeld, Asst. Network Manager	37-422A	3-3322
Geoffrey B. Crew, Chairman, Computer Committee	37-515	3-3789
William F. Marlar Lounge		
Arlyn Hertz, Reservation Coordinator	37-241	3-1456
CSR Reading Room		
Michael Richard, Librarian	37-582	3-3746
HOURS: Monday through Friday, 1:00 p.m. - 5:00 p.m.		

---

except where noted, CSR FAX number is 253-0861

GROUPS WITHIN CSR

	ROOM	EXTENSION
ACIS Project Office		
William F. Mayer, Program Manager	NE80-6073	3-5807
Deborah Gage, Senior Secretary	NE80-6044	3-0228
FAX NUMBER: 253-8084		
AXAF Science Center		
Claude R. Canizares, Head of MIT Collaboration	37-241	3-7501
Arlyn Hertz, Administrative Secretary	37-241	3-1456
CCD Laboratory		
George R. Ricker, Head	37-535	3-7532
Rosemary Hanlon, Administrative Assistant	37-538	3-8433
Laboratory/Common Room	37-518	3-7466
FAX NUMBER: 258-6921		
Gravity And Cosmology Research/LIGO Project		
Rainer Weiss, Head	20F-102	3-3527
Michael Richard, Senior Secretary	20B-145	3-4824
Will Plummer, Secretary	20B-145	3-4824
FAX NUMBER: 253-7014		
HETE Project Office		
Eugene B. Galton, Program Manager	37-511	3-9847
Rosemary Hanlon, Administrative Assistant	37-538	3-8433
HETG Project Office		
Eugene Galton, Program Manager	37-415	3-7294
Kimberly Farrell, Senior Secretary	37-607	3-1736
Man-Vehicle Laboratory		
Charles M. Oman, Head	37-211	3-7508
Secretary	37-219	3-7805

---

except where noted, CSR FAX number is 253-0861

GROUPS WITHIN CSR

		ROOM	EXTENSION
Space Plasma Group			
John W. Belcher, Co-Head		37-695	3-4385
Alan J. Lazarus, Co-Head		37-687	3-4280
Beverly Linton, Senior Secretary		37-675	3-3769
Michigan-Dartmouth-MIT Observatory			
Richard Binzel, Head of MIT Collaboration		54-418	3-6486
Secretary		6-216	3-3718
MDM Main Number:	(602) 322-3360		
MDM Fax Number:	**(602) 322-3364**		
Theoretical Astrophysics Group			
Secretary		6-216	3-3718
FAX NUMBER: 253-9798			
Theoretical Geo/Cosmo Plasma Physics			
Tom T.S. Chang, Head		37-261	3-7523
Mary Terhune, Senior Secretary		37-271	3-7527
XTE Project Office			
William F. Mayer, Program Manager		NE80-6043	3-7552
FAX NUMBER: 253-8084			
Space Microstructures Laboratory			
Mark S. Schattenburg, Head		37-421	3-3180
Kimberly Farrell, Senior Secretary		37-607	3-1736
Laboratory		37-484	8-8615
Clean Rooms	Holography Lab	37-482	8-8622
	Main Lab	37-486	8-8621

---

except where noted, CSR FAX number is 253-0861