BAROMETRIC PRESSURE TRANSIENT TESTING APPLICATIONS
AT THE NEVADA TEST SITE: FORMATION PERMEABILITY ANALYSIS

FINAL REPORT

UCRL--15699

DE85 016078

Prepared by:

Jonathan M. Hanson
Manager, Applied Geophysics

Terra Tek Research
University Research Park
420 Wakara Way
Salt Lake City, Utah 84108

For:

University of California
Lawrence Livermore National Laboratory
Livermore, California 94550

Contract No. 5054005

Attn: Dr. Joseph Hearst

December, 1984

ACKNOWLEDGEMENTS

This report is a summary of work initiated by the author while affiliated with the Earth Sciences Department of the University of California, Lawrence Livermore National Laboratory (LLNL), and the subsequent extension of this work while at Terra Tek Research, Salt Lake City. The extension of this work, which represents the bulk of this report, was carried out under a contract with LLNL through the Earth Sciences Department Containment Program.

The author thanks Dr. Joseph Hearst, LLNL, for help in carrying out the multi-layer data inversions on the CDC 7600 and CRAY computers at LLNL. The author also wishes to thank Mr. Tod Schrauf, Terra Tek Research, for help in code development, and Messrs. Bret Hamatake and Rob Luzitano, both of Terra Tek, for help in data entry.

Finally, the author wishes to dedicate this work to the late Dr. Frank Morrison, Containment Program Leader at LLNL, who played a key role in establishing this research project.

DISCLAIMER

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

TABLE OF CONTENTS

	Page
List of Figures	v
List of Tables	xiii
Chapter I - Introduction and Background	1
Chapter II - Analytic and Semi-Analytic Models for 1-D (Vertical) and 2-D (Vertical and Radial) Flow	5
2.1 - One-Dimensional Flow Models in a Homogeneous and Isotropic Formation	5
2.1.1 - 1-D (Vertical) Flow in a Homogeneous and Isotropic Half-Space - Time Domain	5
2.1.2 - 1-D (Vertical) Flow in a Homogeneous and Isotropic Half Space - Frequency Domain	9
2.2 - One Dimensional Flow Models in a Layered Half-Space	10
2.2.1 - 1-D (Vertical) Flow in a Layered Half- Space - Time Domain	10
2.2.2 - 1-D (Vertical) Flow in a Layered Half- Space - Frequency Domain	15
2.3 - Two Dimensional Models in a Homogeneous and Isotropic Half-Space	15
2.3.1 - Semi-Analytic Model for a Partially Cased Hole	15
2.4 - The Need for a More General Model, and More Sophisticated Methods for Using the Model to Interpret Barometric Data	28

TABLE OF CONTENTS (CONT'D)

	Page
Chapter III - A Numerical Model for 2-D Flow in a Layered Half-Space	31
Chapter IV - Examples of Forward Models	35
Chapter V - Analysis of Barometric Pressure Data From U2fe, U19ac, and U20ai at the Nevada Test Site	51
5.1 - Analysis of Data Taken at NTS Hole U2fe	51
5.2 - Analysis of Data Taken at NTS Hole U19ac	68
5.3 - Barometric Tide Contamination Effects	115
5.4 - Analysis of Data Taken at NTS Hole U20ai	135
Chapter VI - Reevaluation of Previous Air Injection Tests	143
6.1 - Air Injection Tests at U10am-5 - July, 1971 ..	143
6.2 - Air Injection Tests at NTS Hole U9-ITS-eIX30.5	147
Chapter VII - Discussion and Conclusions	151
Appendix - Computer Program Listings	161

LIST OF FIGURES

<u>Figure</u>	<u>Page</u>
2.1 Geometry of open-hole barometric pressure transient testing. Partially cased hole is sealed at the surface. Both surface and in-hole pressure variations are monitored.....	6
2.2 Generalized diagram of a partially cased hole intersecting a layered half-space.....	11
2.3 Schematic of principle flow directions about a partially cased hole subject to an increased surface pressure.....	16
2.4 Radial pressure gradient evaluated at the hole face over the uncased portion of the open hole. t = 10 hrs.....	22
2.5 Radial pressure gradient evaluated at the hole face over the uncased portion of the open hole. t = 100 hrs.....	23
2.6 Vertical pressure profile from the surface to the no-flow boundary at z = 1000 m. 1-D vertical (no-hole) solution is super-imposed on 2-D semi-analytic solution. t=10 hrs.....	25
2.7 Vertical pressure profile from the surface to the no-flow boundary at z = 1000m. 1-D vertical (no-hole) solution is super-imposed on 2-D semi-analytic solution. t=100 hrs.....	26
2.8 Cavity pressure (change) history due to a unit increase in pressure at the surface at t=0. Solid line is 2-D semi-analytic solution. Dashed line is 1-D (vertically averaged) solution....	27
3.1 Computational flow chart for nonlinear least-squares data inversion.....	33
4.1 "Model A". Partially cased hole intersecting a homogeneous and isotropic half-space.....	36
4.2 Computed cavity pressure change vs. time for the finite difference 2-D flow model (solid line) and 1-D (vertically averaged) model (dashed line) - Model A.....	37
4.3 Radial pressure gradient evaluated at the wellface using the finite-difference 2-D flow model after 100 hrs.- Model A.....	38

4.4	Pressure change along the cavity axis (dashed line) and at the outer radial no-flow boundary (solid line) computed using the finite-difference 2-D flow model after 100 hrs. Superimposed on the outer boundary profile is the semi-analytic 1-D solution as a check.....	39
4.5	"Model B". Partially cased hole intersecting a layered half-space.....	40
4.6	Computed cavity pressure change vs. time for the finite difference 2-D model (solid line) and the 1-D (vertically averaged) model (dashed line) - Model B.....	41
4.7	Radial pressure gradient evaluated at the wellface using the finite-difference 2-D flow model after 100 hrs. - Model B.....	43
4.8	Pressure change along cavity axis (dashed line) and along the outer radial no-flow boundary (solid line) computed using the finite-difference 2-D flow model. Superimposed on the outer boundary profile is the semi-analytic 1-D solution as a check.....	44
4.9	"Model C". Partially cased hole intersecting a layered half-space.....	45
4.10	Computed cavity pressure change vs. time for the finite difference 2-D model (solid line) and the 1-D (vertically averaged) model (dashed line) - Model C.....	46
4.11	Radial pressure gradient evaluated at the wellface using the finite-difference 2-D flow model after 100 hrs. - Model C.....	47
4.12	Pressure change along cavity axis (dashed line) and along outer radial no-flow boundary (solid line) computed using the finite-difference 2-D flow model. Superimposed on the outer boundary profile is the semi-analytic 1-D solution as a check.....	48
4.13	Radial pressure gradients evaluated at the wellface after 100 hrs. for Models A,B, and C.....	49
4.14	Change in cavity axis pressures after 100 hrs. for Models A,B, and C.....	50
5.1	Map showing Area locations at the Nevada Test Site...	52
5.2	Pre-shot location map of emplacement hole U2fe. (from Howard, 1982).....	53

5.3	Geologic cross-section through emplacement hole U2fe. (from Howard, 1982).....	54
5.4	Surface sensor #1 pressure record - U2fe.....	56
5.5	Surface sensor #2 pressure record - U2fe.....	57
5.6	In-hole sensor pressure record - U2fe.....	58
5.7	Amplitude spectrum of surface sensor #1 - U2fe.....	59
5.8	Amplitude spectrum of surface sensor #2 - U2fe.....	60
5.9	Amplitude spectrum of in-hole sensor - U2fe.....	61
5.10	"Best-fit" of predicted in-hole pressure (dotted) using surface sensor #1 data and 1-D (vertically averaged) model, superimposed on measured in-hole pressure data (solid) - U2fe.....	62
5.11	"Best-fit" of predicted in-hole pressure (dotted) using surface sensor #2 data and 1-D (vertically averaged) model, superimposed on measured in-hole pressure data (solid) - U2fe.....	63
5.12	Data inversion results on one, two, and three layer models using the finite-difference 2-D flow model - U2fe. Note that the three-layer model was terminated after one iteration, and the minimum in chi-square was not yet reached.....	67
5.13	Pahute Mesa drill hole location map, showing location (see arrow) of hole U19ac. (from Howard, 1984).....	69
5.14	Geologic cross section through U19ac. (from Howard, 1984).....	70
5.15	NE-SW stratigraphic correlation at U19ac. (from Howard, 1984).....	72
5.16	Surface sensor #1 pressure record - File #1, U19ac...	73
5.17	Surface sensor #2 pressure record - File #1, U19ac...	74
5.18	In-hole sensor pressure record - File #1, U19ac.....	75
5.19	Surface sensor #1 pressure record - File #2, U19ac...	76
5.20	Surface sensor #2 pressure record - File #2, U19ac...	77
5.21	In-hole sensor pressure record - File #2, U19ac.....	78
5.22	Surface sensor #1 pressure record - File #3, U19ac...	79
5.23	Surface sensor #2 pressure record - File #3, U19ac...	80

5.24	In-hole sensor pressure record - File #3, U19ac.....	81
5.25	Surface sensor #1 pressure record - File #4, U19ac...	82
5.26	Surface sensor #2 pressure record - File #4, U19ac...	83
5.27	In-hole sensor pressure record - File #4, U19ac.....	84
5.28	Surface sensor #1 pressure record - File #5, U19ac...	85
5.29	Surface sensor #2 pressure record - File #5, U19ac...	86
5.30	In-hole sensor pressure record - File #5, U19ac.....	87
5.31	Amplitude spectrum, surface sensor #1, File #1, U19ac.....	89
5.32	Amplitude spectrum, surface sensor #2, File #1, U19ac.....	90
5.33	Amplitude spectrum, in-hole sensor, File #1, U19ac...	91
5.34	Amplitude spectrum, surface sensor #1, File #2, U19ac.....	92
5.35	Amplitude spectrum, surface sensor #2, File #2, U19ac.....	93
5.36	Amplitude spectrum, in-hole sensor, File #2, U19ac...	94
5.37	Amplitude spectrum, surface sensor #1, File #3, U19ac.....	95
5.38	Amplitude spectrum, surface sensor #2, File #3, U19ac.....	96
5.39	Amplitude spectrum, in-hole sensor, File #3, U19ac...	97
5.40	Amplitude spectrum, surface sensor #1, File #4, U19ac.....	98
5.41	Amplitude spectrum, surface sensor #2, File #4, U19ac.....	99
5.42	Amplitude spectrum, in-hole sensor, File #4, U19ac...	100
5.43	Amplitude spectrum, surface sensor #1, File #5, U19ac.....	101
5.44	Amplitude spectrum, surface sensor #2, File #5, U19ac.....	102
5.45	Amplitude spectrum, in-hole sensor, File #5, U19ac...	103

5.46	"Best-fit" of predicted in-hole pressure (dotted) using surface sensor #1, File #1, data and 1-D (vertically averaged) model, superimposed on measured in-hole pressure data (solid) - U19ac.....	104
5.47	"Best-fit" of predicted in-hole pressure (dotted) using surface sensor #2, File #1, data and 1-D (vertically averaged) model, superimposed on measured in-hole pressure data (solid) - U19ac.....	105
5.48	"Best-fit" of predicted in-hole pressure (dotted) using surface sensor #1, File #2, data and 1-D (vertically averaged) model, superimposed on measured in-hole pressure data (solid) - U19ac.....	106
5.49	"Best-fit" of predicted in-hole pressure (dotted) using surface sensor #2, File #2, data and 1-D (vertically averaged) model, superimposed on measured in-hole pressure data (solid) - U19ac.....	107
5.50	"Best-fit" of predicted in-hole pressure (dotted) using surface sensor #1, File #3, data and 1-D (vertically averaged) model, superimposed on measured in-hole pressure data (solid) - U19ac.....	108
5.51	"Best-fit" of predicted in-hole pressure (dotted) using surface sensor #2, File #3, data and 1-D (vertically averaged) model, superimposed on measured in-hole pressure data (solid) - U19ac.....	109
5.52	"Best-fit" of predicted in-hole pressure (dotted) using surface sensor #1, File #4, data and 1-D (vertically averaged) model, superimposed on measured in-hole pressure data (solid) - U19ac.....	110
5.53	"Best-fit" of predicted in-hole pressure (dotted) using surface sensor #2, File #4, data and 1-D (vertically averaged) model, superimposed on measured in-hole pressure data (solid) - U19ac.....	111
5.54	"Best-fit" of predicted in-hole pressure (dotted) using surface sensor #1, File #5, data and 1-D (vertically averaged) model, superimposed on measured in-hole pressure data (solid) - U19ac.....	112
5.55	"Best-fit" of predicted in-hole pressure (dotted) using surface sensor #2, File #5, data and 1-D (vertically averaged) model, superimposed on measured in-hole pressure data (solid) - U19ac.....	113
5.56	Zero-phase low-pass filter in the time domain.....	116
5.57	Zero-phase low-pass filter in the frequency domain..	117

5.58	90% confidence envelopes of principle tidal constituents, surface sensor #1, File #5 - U19ac....	119
5.59	90% confidence envelopes of principle tidal constituents, measured in-hole pressure, File #5 - U19ac.....	120
5.60	90% confidence envelopes of principle tidal constituents, predicted in-hole pressure, File #5 - U19ac.....	121
5.61	"Best-fit" of predicted in-hole pressure (dotted) using filtered surface sensor #1 data and 1-D (vertically averaged) model, superimposed on measured, filtered in-hole pressure data (solid) - U2fe.....	122
5.62	"Best-fit" of predicted in-hole pressure (dotted) using filtered surface sensor #2 data and 1-D (vertically averaged) model, superimposed on measured, filtered in-hole pressure data (solid) - U2fe.....	123
5.63	"Best-fit" of predicted in-hole pressure (dotted) using filtered surface sensor #1, File #1 data and 1-D (vertically averaged) model, superimposed on measured, filtered in-hole pressure data (solid) - U19ac.....	124
5.64	"Best-fit" of predicted in-hole pressure (dotted) using filtered surface sensor #2, File #1 data and 1-D (vertically averaged) model, superimposed on measured, filtered in-hole pressure data (solid) - U19ac.....	125
5.65	"Best-fit" of predicted in-hole pressure (dotted) using filtered surface sensor #1, File #2 data and 1-D (vertically averaged) model, superimposed on measured, filtered in-hole pressure data (solid) - U19ac.....	126
5.66	"Best-fit" of predicted in-hole pressure (dotted) using filtered surface sensor #2, File #2 data and 1-D (vertically averaged) model, superimposed on measured, filtered in-hole pressure data (solid) - U19ac.....	127
5.67	"Best-fit" of predicted in-hole pressure (dotted) using filtered surface sensor #1, File #3 data and 1-D (vertically averaged) model, superimposed on measured, filtered in-hole pressure data (solid) - U19ac.....	128
5.68	"Best-fit" of predicted in-hole pressure (dotted) using filtered surface sensor #2, File #3 data	

and 1-D (vertically averaged) model, superimposed on measured, filtered in-hole pressure data (solid) - U19ac.....	129
5.69 "Best-fit" of predicted in-hole pressure (dotted) using filtered surface sensor #1, File #4 data and 1-D (vertically averaged) model, superimposed on measured, filtered in-hole pressure data (solid) - U19ac.....	130
5.70 "Best-fit" of predicted in-hole pressure (dotted) using filtered surface sensor #2, File #4 data and 1-D (vertically averaged) model, superimposed on measured, filtered in-hole pressure data (solid) - U19ac.....	131
5.71 "Best-fit" of predicted in-hole pressure (dotted) using filtered surface sensor #1, File #5 data and 1-D (vertically averaged) model, superimposed on measured, filtered in-hole pressure data (solid) - U19ac.....	132
5.72 "Best-fit" of predicted in-hole pressure (dotted) using filtered surface sensor #2, File #5 data and 1-D (vertically averaged) model, superimposed on measured, filtered in-hole pressure data (solid) - U19ac.....	133
5.73 Data inversion results on a three layer model using the finite-difference 2-D flow model - U19ac.....	136
5.74 Surface sensor #1 pressure record - U20a1.....	137
5.75 Surface sensor #2 pressure record - U20a1.....	138
5.76 In-hole sensor pressure record - U20a1.....	129
5.77 "Best-fit" of predicted in-hole pressure (dotted) using surface sensor #1 and 1-D (vertically averaged) model, superimposed on measured in- hole pressure data (solid) - U20a1.....	140
5.78 "Best-fit" of predicted in-hole pressure (dotted) using surface sensor #2 and 1-D (vertically averaged) model, superimposed on measured in- hole pressure data (solid) - U20a1.....	141
6.1. Hole geometry at U10am-5.....	144
6.2. Geologic cross section through U9-ITS-XY31.....	148

LIST OF TABLES

Table	Page
5.1 Description of Lithology at U2fe	55
5.2 Computed Diffusivities at NTS Hole U2fe	65
5.3 Computed Permeabilities at NTS Hole U2fe	65
5.4 Description of Lithology at U19ac	71
5.5 Start and Stop Times for U19ac Data Sets	68
5.6 Computed Diffusivities for NTS Hole U19ac	88
5.7 Precipitation Record at Hole U19ac	113
5.8 Computed Permeabilities at NTS Hole U19ac	114
5.9 Atmospheric Tidal Constituents	118
5.10 Recomputed Diffusivities at NTS Hole U2fe	134
5.11 Recomputed Diffusivities at NTS Hole U19ac	134
5.12 Computed Diffusivity at U20ai	135
5.13 Computed Permeabilities for U20ai	142
6.1 Air Injection Tests at U10am-5	143
6.2 Reevaluated Permeabilities at NTS Hole U10am-5	146
6.3 Air Injection Tests at NTS Hole U9-ITS-eXY30.5	147
6.4 Averaged Permeabilities at Hole U9-ITS-eXY30.5	149
6.5 "Deconvolved" Formation Permeabilities at NTS Hole U9-ITS-eXY30.5	150

CHAPTER I

INTRODUCTION AND BACKGROUND

The evaluation of gas permeability of the rock surrounding emplacement holes at the Nevada Test Site (NTS) is an extremely important issue that probably has not received enough attention in the past. The importance of this parameter lies in the fact that its magnitude will play a major role in establishing the likelihood of radioactive gas transport from the shot point to the surface during or long after an underground nuclear test. This leakage can occur in several ways: (a) simple molecular diffusion along permeable pathways, (b) convective diffusion either by atmospheric pumping or heat released at the shot point, or (3) the generation of steam-driven fractures propagated away from the shot point due to the heat generation at the shot point. Permeability plays a dominant role in the latter in that the distance the steam drive fractures propagate depends strongly on pressure leak-off from the fracture to the formation during propagation.

Several investigations have been carried out in the past to evaluate undisturbed (in-situ) formation permeability at the NTS. These include conventional air-injection tests (Snoeberger, et al., 1971), the analysis of barometric pressure fluctuations (Rozsa, et al., 1974), and ground water pump tests in deep wells on the Pahute Mesa (Blankennagel and Weir, 1973). In spite of these investigations, there still exist heated debates on the question of formation permeability at the NTS.

There have been differing opinions on the magnitude of permeability at the Nevada Test Site. Strongly held opinions range from "certainly greater than a Darcy" to "probably less than a millidarcy" for the alluvium. Some of this uncertainty may derive from the ever-present problem of scaling core permeability tests to field dimensions. Other ambiguities arise from assuming that the permeability measured at one site should necessarily correlate with the permeability at a nearby site or that the permeability of a geologic layer is necessarily the same for the layer throughout the site. This correlation is probably reasonable for the alluvium, but is much more questionable for fractured volcanics. Finally, there simply has not been enough in-situ permeability testing at the Nevada Test Site to establish a sufficient data base required to answer either permeability magnitude or spatial variability questions. "Benchmark" permeability values seem to have been set based on a very limited number of field experiments. It is the author's opinion that insufficient testing has been carried out to set benchmarks, or even if it makes sense to define a benchmark, considering that most holes at the Nevada Test Site intersect formations volcanic in nature.

An integral part of this report is the evaluation of past investigations of permeability at the Nevada Test Site. Due to the critical nature of this parameter regarding containment, previous investigations were critiqued in a rigorous fashion. The following discussion is presented to set the framework from which the present uncertainty in gas permeability at the Nevada Test Site can be discussed, and how this uncertainty can be overcome. We have made every attempt to obtain a complete account of previous work on the evaluation of in-situ unsaturated zone permeability at the Nevada Test Site. We hope that the work cited in the following fairly represents past efforts on this topic, although we were surprised at the limited number of documents we found in this search.

The conventional air injection tests, carried out at the exploratory hole U9-ITS-eiy30.5, yielded values of 8.2 Darcies for the alluvium, 12.9 Darcies for vitric tuffs, 4.0 Darcies for the Grouse Canyon ashfall tuff, and 2.5 Darcies for undifferentiated tuffs and tuffaceous sandstones (Snoeberger et al, 1971). Even though these were pressure transient tests consisting of "relatively short times - approximately 20 minutes, sometimes less - for pressure buildup", the interpretation of the data assumed that the pressure distribution had reached steady state. Another set of air-injection tests were carried out on NTS hole U10am-5 in 1971. Again using the steady state solution, values of between 8 and 16 Darcies for the alluvium were obtained (Morris, et al, 1971; Rozsa, et al., 1972). Chapter VI presents a reinterpretation of the air-injection data recorded at both of those sites using a transient model.

The one documented analysis of barometric pressure fluctuations, carried out for the purpose of evaluating formation permeability, was at hole U9ITS-eV-29#2 (Rozsa, et al., 1974). This hole penetrated the surface alluvium, ashfall and bedded tuffs, and partially penetrated Paleozoic limestones. Several models were used to interpret the data, including 1-D (vertical), and 2-D (radial and vertical) flow in a homogeneous and isotropic formation. Values of permeability ranged between 0.8 Darcies (1-D vertical model evaluated at the bottom of the casing) and 60 Darcies (2-D model). The 2-D model gave the best fit to the data. The authors concluded that "since previous work shows the permeability of surface alluvium to be about 8 Darcies, both of the above values are suspect". This is what is meant by "benchmarking" referred to previously.

A suite of pump tests were carried out on packed-off intervals of a number of exploratory holes at Pahute Mesa (Plankennagel and Weir, 1973). The test interval depths ranged between 1000 and 4300 feet below the surface at locations corresponding to ash fall tuffs, welded tuffs, and rhyolite flows. All tests were carried out using conventional pump testing method below the water table. Computed transmissivities ranged between 460 Darcy-feet and 7780 Darcy-feet. Based on the packed-off section interval thickness, these transmissivity values correspond to a permeability range of roughly 3 to 50 Darcies. Permeability in

these formations is attributed primarily to fractures - the interstitial matrix permeability is small in comparison.

It is clear from the above that, using a variety of different field permeability testing techniques, measured in-situ permeability at the NTS has been consistently found to be greater than a few Darcies and as high as many tens of Darcies. In particular, the alluvium permeability seems to roughly range between 1 and 50 Darcies.

The field data acquisition procedures and data analysis methods for conventional pump testing, including the air injection and groundwater withdrawal tests discussed above, have had extensive use and verification in both the petroleum industry and in groundwater analysis. Unfortunately, these techniques are either very expensive or impractical for permeability testing above the water table at the NTS. This is particularly true of large-diameter emplacement holes. Air injection testing requires a substantial surface and downhole equipment inventory including surface compressors, surface pressure regulation manifolds, wellhead piping, and possibly downhole packers for small-diameter exploratory holes. It is impractical, if not impossible, to pack off sections of large-diameter emplacement holes for permeability testing. Air injection testing on cased and perforated slim holes or larger holes with injection into a packed-off interval, on the other hand, presents a very useful tool to critically establish very good values of spatially resolved permeability. With a smaller diameter hole, or packed-off open or perforated down-hole zones in larger diameter holes, surface air injection equipment is minimized. Typically, however, "holes of opportunity" which have satisfied none of the above criteria, either in hole size or test geometry, have been used for air injection.

Conventional pump testing interpretation models commonly assume a purely radial (cylindrical) flow away from the wellbore. This is usually a good approximation for petroleum reservoirs where shale breaks, silt stringers, or other stratigraphic flow barriers constrain flow to be radial. This is not true of the NTS alluvium, where there are only subtle lithologic gradations as a function of depth. The same case can be made for pressure testing in the volcanics of the NTS, where fracture flow must be highly anisotropic. Thus, in order to properly interpret test data, a 2-D (vertical and radial) flow model, with some allowance for permeability anisotropy, must be used. This degree of sophistication in air injection test interpretation has not been met in past analyses of tests at the NTS (Snoeberger, et al., 1971). It is strongly recommended that, if air injection testing be seriously considered in the future, data interpretation methods address these concerns.

By far the most inexpensive method of formation permeability testing at the NTS is the barometric pressure transient method. This method has been used for estimating permeability surrounding partially cased emplacement holes at the NTS (Rosen, et al., 1974; Hanson, 1982) and has also been used for estimating

vertical permeability of unsaturated soils (Weeks, 1978) and in unsaturated formations overlying unconfined aquifers (Weeks, 1979). In addition to formation permeability, this technique has been used extensively for estimating nuclear chimney permeability at the NTS (Morris and Snoeberger, 1971; Snoeberger, et al., 1972; Snoeberger, et al., 1973; Snoeberger, et al., 1974a,b; Rozsa, et al., 1974; and Rozsa, et al., 1975).

The technique, as applied in this report, assumes that a quartz pressure gauge is suspended in a partially cased hole. The top of the casing is sealed so that there is no direct communication with the atmosphere. By comparing the measured in-hole pressure record with the surface barometric pressure record using an appropriate pressure diffusion model, the value of in-situ permeability of the formation intersected by the hole can be computed. The technique has not been well established. This is due principally to the fact that it is a relatively new method and also that the interpretation methods are not well developed. We hope, in this report, to present a solid foundation for interpreting this data, apply these techniques to data taken at the Nevada Test Site, and discuss both the advantages and disadvantages of this technique in light of the computed results.

CHAPTER II

ANALYTIC AND SEMI-ANALYTIC MODELS FOR 1-D (VERTICAL) AND 2-D (VERTICAL AND RADIAL) FLOW

Evaluation of permeability using the barometric pressure method can be carried out using two approaches. The first method, useful primarily in moderately shallow slim holes, assumes that the down-hole pressure sensor has been grouted in, but still has access to the adjacent formation. Under this restriction, the hole is "invisible" as regards its effect on the flow field. If the flow parameters (permeability and porosity) are uniform (homogeneous and isotropic) in the half-space, the flow field is 1-D (vertical) and therefore, the pressure response at the sensor can be adequately approximated by a 1-D model. Also, if the half-space is layered, each layer being homogeneous and isotropic in its flow parameters, the model is again 1-D. Weeks (1978) has used this geometry to evaluate vertical permeability to air in the unsaturated zones in soils.

The second technique is to suspend the down-hole pressure sensor in a partially cased, but otherwise open, hole. Figure 2.1 shows this geometry. In this case, the hole acts as an almost infinitely good conductor for redistributing mass from regions of high formation pressure to regions of low formation pressure. Because of this, a significant component of radial flow to and from the wellbore will be present, and a 1-D model can no longer adequately represent the flow field.

In the following, analytic and semi-analytic solutions will be derived for the cases of: (a) 1-D (vertical) flow in a homogeneous and isotropic half-space, (b) 1-D (vertical) flow in a layered half-space, and (c) 2-D (vertical and radial) flow around a partially cased finite-length vertical hole in a homogeneous and isotropic half-space.

2.1 ONE-DIMENSIONAL FLOW MODELS IN A HOMOGENEOUS AND ISOTROPIC FORMATION

2.1.1 1-D (Vertical) Flow In a Homogeneous and Isotropic Half-Space - Time Domain

The mathematical statement of this problem is as follows:

Figure 2.1. Geometry of open-hole barometric pressure transient testing. Partially cased hole is sealed at the surface. Both surface and in-hole pressure variations are monitored.

$$\frac{1}{c} \frac{\partial p}{\partial t} = \frac{\partial^2 p}{\partial z^2}, \quad (2-1)$$

$$z > 0, t > 0$$

$$p(z, t) \Big|_{z=0} = P_s(t) \quad (2-2)$$

$$p(z, t) \Big|_{t=0} = P_0 \quad (2-3)$$

$$p(z, t) \Big|_{z \rightarrow \infty} = 0 \quad (2-4)$$

Taking the Laplace transform with respect to time of equation (2-1), one obtains:

$$\frac{\partial^2 \hat{p}}{\partial z^2} - \frac{s}{c} \hat{p} = -\frac{1}{c} P_0 \quad (2-5)$$

The general solution of this differential equation, prior to imposing boundary condition constraints, is:

$$\hat{p}(z, s) = A e^{\sqrt{\frac{s}{c}} z} + B e^{-\sqrt{\frac{s}{c}} z} + \frac{P_0}{s} \quad (2-6)$$

If we now impose the boundary constraints given by eqs. (2-2) and (2-4), we obtain the solution (in Laplace space):

$$\hat{p}(z,s) = \left(\hat{P}_s - \frac{P_0}{s}\right) e^{-\sqrt{\frac{s}{c}} z} + \frac{P_0}{s} \quad (2-7)$$

Using the convolution theorem for products of Laplace transforms (Abramowitz and Stegun, 1964), one can obtain an exact expression for the solution in the time domain. This is given by:

$$\begin{aligned} p(z,t) &= \int_0^t [P_s(\tau) - P_0] G(z;t-\tau) d\tau + P_0 \\ &= \int_0^t [P_s(t-\tau) - P_0] G(z;\tau) d\tau + P_0 \end{aligned} \quad (2-8)$$

Where the Green's function kernel is given by:

$$G(z;t-\tau) = \frac{z}{2\sqrt{\pi c(t-\tau)^3}} e^{-\frac{z^2}{4c(t-\tau)}} \quad (2-9)$$

Thus, by using equation (2-9), the pressure response $P_s(t)$ at sensor depth z can be calculated as a function of time for an arbitrary surface barometric pressure $P(t)$.

This is a very compact solution, and represents a model that is likely over-simplified for all but the most ideal field situations. However, the solution does reflect a characteristic common to all diffusion processes: the pressure at time t depends strongly on the history of the surface pressure. If surface and down-hole pressure were monitored over the time period (t_1, t_2) , and this data were to be used to extract the gas diffusivity, C , of the formation, some care must be taken in the model-data matching procedure. Down-hole data at time t_1 is negligibly affected by the surface pressure at the same time. Therefore, the model should be fit to down-hole data for a time span (t_1, t_2) , where $t_1 < t_3 < t_2$. Unfortunately, t_3 depends on the parameter of interest, namely the gas diffusivity C .

In order to circumvent some of these problems, it is useful to cast the downhole pressure response function in the frequency domain. To a reasonable approximation, barometric pressure is a stationary time series. This means that its spectrum does not vary with time. As a result, matching a pressure response model to data is much more straight forward.

2.1.2 1-D (Vertical) Flow in a Homogeneous and Isotropic Half Space - (Frequency Domain)

The statement of this problem is identical to that of section 2.1.1, with the exception that the surface pressure signal is assumed to have the form:

$$p_s(t) = P_s e^{i\omega t} \quad (2-10)$$

Consequently, the downhole pressure must have the form:

$$p(z,t) = N(z) e^{i\omega t} \quad (2-11)$$

where $N(z)$ is a complex number containing pressure amplitude and phase information. Using these definitions, the diffusion equation (2-1) can be transformed into the following:

$$\frac{\partial^2 N}{\partial z^2} - \frac{i\omega}{c} N = 0 \quad (2-12)$$

Invoking the boundary condition given by eq. (2-4), the solution to eq. (2-12) is found to be:

$$\begin{aligned} p(z,\omega) &= P_s e^{-\sqrt{\frac{i\omega}{c}} z} \\ &= P_s e^{\sqrt{\frac{\omega}{2c}} z} e^{-i\sqrt{\frac{\omega}{2c}} z} \end{aligned} \quad (2-13)$$

One should note the similarity of this solution with that for

the time domain given by eq. (2-7). Setting the initial formation gas pressure P_0 to zero, we see that if the parameter $i\omega$ is substituted for s in eq. (2-7), we obtain eq. (2-13). This reflects the well known correspondence of Laplace transforms and Fourier transforms under a rotation of 90 degrees in the complex plane. This correspondence will be used to advantage in the next section.

2.2 ONE DIMENSIONAL FLOW MODELS IN A LAYERED HALF-SPACE

2.2.1 1-D (Vertical) Flow in a Layered Half-Space (Time Domain)

The solution to the 1-D flow layered half-space problem is similar to the solution given for the homogeneous half-space by eq. (2-6), with the exception that now, we have two additional boundary condition constraints. These constraints derive from the requirement of pressure continuity and mass flux continuity across the layer interfaces. Figure 2.2 shows the layered half-space geometry for n layers intersected with a partially cased hole. The 1-D layered half-space solution will be valid for radial distances far from the hole such that the only flow component is vertical. The mathematical statement of the problem is given as follows:

$$\frac{1}{C_i} \frac{\partial p_i}{\partial t} = \frac{\partial^2 p_i}{\partial z^2}, \quad z_{i-1} < z < z_i, \quad i = 1, 2, \dots, n \quad (2-14)$$

$$p_i(z, t) \Big|_{z=0} = P_g(t) \quad (2-15)$$

$$p_i(z, t) \Big|_{t=0} = P_0, \quad i = 1, 2, \dots, n \quad (2-16)$$

$$p_i(z, t) \Big|_{z \rightarrow \infty} = 0 \quad (2-17)$$

Figure 2.2. Generalized diagram of a partially cased hole intersecting a layered half-space.

$$p_i(z_i, t) = p_{i+1}(z_i, t), \quad i = 1, 2, \dots, n-1 \quad (2-18)$$

$$k_i \left. \frac{\partial p_i}{\partial z} \right|_{z=z_i} = k_{i+1} \left. \frac{\partial p_{i+1}}{\partial z} \right|_{z=z_i}, \quad i = 1, 2, \dots, n \quad (2-19)$$

Eqs. (2-18) and (2-19) represent the pressure and mass flux continuity constraints at the layer interfaces. Casting eqs. (2-14) through (2-19) in Laplace transform space, one can obtain a set of $2n-2$ equations representing the $2n-2$ unknowns. The latter are the set of layer coefficients $\{A(1), B(1)\}, \{A(2), B(2)\}, \{A(3), B(3)\}, \dots, \{A(n), B(n)\}$, where each set of brackets $\{\}$ represent a single layer in the n -layer half-space. The general solution for the i th layer (in Laplace transform space) is given by:

$$\hat{p}_i(z, s) = A_i e^{\sqrt{\frac{s}{c_i}} z} + B_i e^{-\sqrt{\frac{s}{c_i}} z} + \frac{p_0}{s} \quad (2-20)$$

$$z_{i-1} < z < z_i$$

$$i = 1, 2, \dots, n$$

Note that $B(n)$ is zero due to the requirement that the solution go to zero as depth goes to infinity. Also, $A(1)$ can be evaluated by noting that:

$$\hat{p}_s = A_1 + B_1 + \frac{p_0}{s} \quad (2-21)$$

Combining eqs. (2-18), (2-19), and (2-20), we can set up an 2×2 matrix that "propagates" the layer coefficients downward. That is, given the coefficients for the i th layer, the coefficients for the

(i+1)th layer can be quickly evaluated with a simple matrix multiplication. With some algebra, it can be shown that:

$$\begin{pmatrix} A_{i+1} \\ B_{i+1} \end{pmatrix} = M \begin{pmatrix} A_i \\ B_i \end{pmatrix} \quad (2-22)$$

where the "propagator matrix", M, is given by:

$$M = X^{-1} Y \quad (2-23)$$

in which

$$X = \begin{pmatrix} x_{11} & x_{12} \\ x_{21} & x_{22} \end{pmatrix}$$

$$Y = \begin{pmatrix} y_{11} & y_{12} \\ y_{21} & y_{22} \end{pmatrix}$$

and

$$x_{11}^i = -k_{i+1} \sqrt{\frac{s}{c_{i+1}}} e^{-\sqrt{\frac{s}{c_{i+1}}} z_i} \quad (2-24)$$

$$x_{12}^i = k_{i+1} \sqrt{\frac{s}{c_{i+1}}} e^{+\sqrt{\frac{s}{c_{i+1}}} z_i}$$

$$x_{21}^i = e^{-\sqrt{\frac{s}{c_{i+1}}} z_i}$$

$$x_{22}^i = e^{+\sqrt{\frac{s}{c_{i+1}}} z_i}$$

$$y_{11}^i = -k_i \sqrt{\frac{s}{c_i}} e^{-\sqrt{\frac{s}{c_i}} z_i}$$

$$y_{12}^i = k_i \sqrt{\frac{s}{c_i}} e^{+\sqrt{\frac{s}{c_i}} z_i}$$

$$y_{21}^i = e^{-\sqrt{\frac{s}{c_i}} z_i}$$

$$y_{22}^i = e^{+\sqrt{\frac{s}{c_i}} z_i}$$

From a computational standpoint, it is most efficient to solve for all of the layer coefficients simultaneously. This is best achieved by noting from eqs. (2-22) and (2-23) and (2-24) that we can write, with some rearrangement:

$$\begin{bmatrix}
 (y_{12}^1 - y_{11}^1) & -x_{11}^1 & -x_{12}^1 & 0 & 0 & 0 & 0 & 0 & 0 \\
 (y_{22}^1 - y_{21}^1) & -x_{21}^1 & -x_{22}^1 & 0 & 0 & 0 & 0 & 0 & 0 \\
 0 & y_{11}^2 & y_{12}^2 & -y_{11}^2 & -y_{12}^2 & 0 & 0 & 0 & 0 \\
 0 & y_{21}^2 & y_{22}^2 & -y_{21}^2 & -y_{22}^2 & 0 & 0 & 0 & 0 \\
 0 & 0 & 0 & 0 & 0 & 0 & y_{11}^n & y_{12}^n & -x_{11}^n \\
 0 & 0 & 0 & 0 & 0 & 0 & y_{21}^n & y_{22}^n & -x_{21}^n
 \end{bmatrix}
 \begin{bmatrix}
 B_1 \\
 A_2 \\
 B_2 \\
 A_3 \\
 B_n \\
 A_{n+1}
 \end{bmatrix}
 \quad (2-25)$$

$$= -\hat{P}_s (y_{11}^1, y_{21}^1, 0, 0, \dots, 0)^T$$

Eq. (2-25) is simply a set of simultaneous equations that can be solved by any number of methods for the unknown coefficients $\{A(i), B(i)\}, i=1, 2, \dots, n$. The $(n+1)$ th layer is a uniform layer of infinite thickness. Note that this system of equations is in Laplace transform space. It would be a tedious, if not impossible, task to analytically invert the solution. We therefore employ a numerical Laplace transform technique (Stehfest, 1969; Crump, 1976).

It should also be pointed out here that the solution derived on the basis of eq. (2-25) is appropriate only for the condition that the Laplace transform of the surface pressure, $P(t)$, is available. In the case of barometric pressure loading at the surface, this will not be the case. One way around this is simply to calculate numerically the Laplace transform of the surface pressure signal. In principal, this is straight forward. However, in practice, the sampling interval in time may preclude obtaining a reasonable precision in calculating the numerical Laplace transform. A better method is to calculate the impulse response of the layered half-space by assuming the surface pressure is a delta function in time. The Laplace transform of a delta function is simply the quantity 1. Once the impulse response is calculated, the pressure response at depth due to an arbitrary surface pressure history is determined by convolving the surface pressure with the impulse response (see also eq. 2-8).

2.2.2 1-D (Vertical) Flow in a Layered Half-Space (Frequency Domain)

One could carry out a similar analysis to obtain a solution in the frequency domain. However, using the previously discussed connection between the Laplace transform and the Fourier transform, we can immediately write the solution. We assume as before a surface pressure varying as $\exp(i\omega t)$. With this assumption, replacing the Laplace parameter s with $i\omega$ in eq. (2-24) will yield a set of simultaneous equations in the frequency domain. Which can be for the unknown coefficients.

2.3 TWO DIMENSIONAL MODELS IN A HOMOGENEOUS AND ISOTROPIC HALF-SPACE

2.3.1 Semi-analytic Model for a Partially Cased Hole

A partially cased hole of given bore radius r_0 and depth $z=L_h$ is cased to a depth $z=L_c$. A "no-flow" boundary at depth $z=L$ simulates either the presence of a water table or, if L be made sufficiently large, a boundary at infinite depth. Figure 2.3 shows the geometry of the problem under consideration. The presence of the open hole forces the flow geometry to have both vertical and radial components.

Figure 2.3. Schematic of principle flow directions about a partially cased hole subject to an increased surface pressure.

This may be seen easier by considering the heat flow analog of a buried finite cylinder with infinite thermal conductivity in a finite thermal conductivity half-space. The top surface and a portion of the cylinder's surface near the top ($z=0, Lc$) are insulated. If the surface of the half-space is heated at $t = 0$, a certain length of time corresponding with the characteristic diffusion time associated with the distance Lc will be required for the energy to access the conducting cylinder. At this time, the cylinder will draw energy from the adjacent formation near $z = Lc$, and will conduct this energy instantaneously along the entire length of the cylinder. At any given time the cylinder will be isothermal. Because the formation in the deeper regions is cooler than the cylinder at these same depths, there will be heat transfer from the cylinder to the formation. Thus, for times greater than the diffusion time associated with the distance Lc , there will be heat transfer from the formation to the cylinder near the surface and heat transfer from the cylinder to the formation near the bottom of the hole.

This translates, for the pressure diffusion problem under consideration, to gas flow into the hole at the top and out of the hole at the bottom of the hole. Thus, the radial component of flow plays a major role in determining the pressure within the hole at any given time.

In the following, we assume that there is no vertical flow out of the bottom of the hole. With this assumption, we derive a solution for the space $(0 < z < L; r > r_0)$, with a mixed boundary condition on $(0 < z < L; r = r_0)$. A conservation of mass constraint connects the pressure in the cavity with the pressure in the formation.

The mathematical statement of the problem is given as follows:

$$\frac{1}{c} \frac{\partial p}{\partial t} = \nabla^2 p, \quad \begin{matrix} 0 < z < L \\ r > r_0 \end{matrix} \quad (2-26)$$

where
$$\nabla^2 = \frac{1}{r} \frac{\partial}{\partial r} \left(r \frac{\partial}{\partial r} \right) + \frac{\partial^2}{\partial z^2}$$

With the following initial and boundary conditions:

$$\left. \frac{\partial p}{\partial r} \right|_{r=r_0} = 0, \quad 0 \leq z \leq L_c, \quad L_h \leq z \leq L \quad (2-27)$$

$$p(z, t)|_{z=0} = p_g(t)$$

$$\left. \frac{\partial p}{\partial z} \right|_{z=L} = 0$$

Subject to the conservation of mass boundary condition at the open hole:

$$\frac{\partial}{\partial t} \int_0^{L_h} p(r, z, t) \Big|_{r=r_0} dz + \frac{2kp_b}{r_0 \mu} \int_{L_c}^{L_h} \left. \frac{\partial p}{\partial r} \right|_{r=r_0} dz = 0 \quad (2-28)$$

Note that in eq. (2-28), we have included the volume of gas within the casing in the mass accumulation term.

The solution of the above set of equations is again carried out in Laplace transform space. Transforming eq. (2-26), we obtain:

$$\nabla^2 \hat{p} - \frac{s}{c} \hat{p} = - \frac{p_0}{s} \quad (2-29)$$

The general solution to eq. (2-29) is taken to have the form of a Fourier series in z :

$$\hat{p}(r,z,s) = \sum_{n=0}^N B_n(r,s) \sin(A_n z) + \hat{p}_h(z,s) \quad (2-30)$$

where p_h is the "no-hole" solution. The coefficients $\{A_n, n=1,2,\dots,N\}$ are chosen such that the boundary condition constraints at the surface $z=0$ and $z=L$ are met. This yields:

$$A_n = (n + 1/2) \pi / L \quad (2-31)$$

Inserting eq. (2-30) into eq. (2-29), one obtains a differential equation for the Fourier coefficients B_n :

$$\frac{\partial^2 B_n}{\partial r^2} + \frac{1}{r} \frac{\partial B_n}{\partial r} - \left(\frac{s}{c} + A_n^2 \right) B_n = 0 \quad (2-32)$$

The general solution to eq. (2-32) that is finite as r gets large is given by:

$$B_n(r,s) = C_n(s) K_0 \left(\sqrt{A_n^2 + \frac{s}{c}} r \right) \quad (2-33)$$

Thus, the general solution for the 2-D pressure field, prior to invoking the conservation of mass constraint at the wellbore, is given by:

$$\hat{p}(r, z, s) = \sum_{n=0}^N C_n(s) K_0(D_n r) \sin(A_n z) + \hat{p}_h(z, s) \quad (2-34)$$

where:

$$D_n = \sqrt{A_n^2 + \frac{s}{c}}$$

Since flow entry and exit to the cylindrical cavity is over a limited depth, we cast the solution in terms of a Neumann Green's function $G(z; z')$ as follows:

$$\hat{p}(r, z, s) = \int_{L_c}^{L_h} G_N(z|z') \left. \frac{\partial p}{\partial r} \right|_{r=r_0} dz' + \hat{p}_h(z, s) \quad (2-35)$$

Note that since the radial pressure gradient at the wellbore is non-zero only $[L_c, L_h]$, the integral need only span this interval, as indicated above. However, eq. (2-35) remains valid if the integral limits span $[0, L]$. This is important in that using this interval allows us to employ the orthogonality properties of the eigenfunctions.

Calculating the radial pressure gradient from eq. (2-34), inserting this gradient in to eq. (2-35), and equating the result with eq. (2-34) allows for defining the Green's function $G(z; z')$, where the orthogonality of the eigenfunctions is used. The result, which is straight forward, but a bit tedious, obtains:

$$G(z; z') = - \frac{2}{L} \sum_{n=0}^N \frac{K_0(D_n r) \sin(A_n z) \sin(A_n z')}{D_n K_1(D_n r_0)} \quad (2-36)$$

We now have a means of obtaining the radial pressure gradient, albeit in Laplace transform space, from which we can calculate the entire pressure field. As discussed in our simple heat conduction analog above, we can consider that, after subtracting the static pressure field (that field due simply to the mean barometric pressure), the pressure within the cavity can be considered constant with depth. This assumption is not all together true since there will be a slight amount of frictional drag at the hole wall due to vertical gas flow in the hole. However, as will be shown in the next chapter, this is a very minor effect. As a result, the conservation of mass boundary condition given by eq. (2-28) can be rewritten as:

$$\hat{R}(s) = - \frac{2kP_b}{r_o \mu s L_h} \int_{L_c}^{L_h} \left. \frac{\partial p}{\partial r} \right|_{r=r_o} dz \quad (2-37)$$

where $\hat{R}(s)$ is the Laplace transform of the cavity pressure. Using this expression along with eq. (2-35), which is evaluated at the wellbore wall, we obtain the Fredholm integral equation:

$$\int_{L_c}^{L_h} \left\{ G(z|z') + \frac{2kP_b}{r_o \mu s L_h} \right\} \left. \frac{\partial p}{\partial r} \right|_{r=r_o} dz' = - \hat{p}_h(z, s) \quad (2-38)$$

Due to the complexity of the kernel of the above equation, an analytic solution is out of the question. Therefore, a numerical approach must be taken. Details of the procedure are not appropriate here. However, a brief outline is in order. Basically, the kernel (the bracketed expression in eq. (2-38)) is cast as a matrix, each element corresponding with a value of (z, z') i.e. a row represents a discretized z axis spanning $[L_c, L_h]$ for a given z' and a column represents a discretized z' axis spanning the same interval for a given z . The right hand side of eq. (2-38) is cast in terms of a column vector spanning the same interval. Thus, the integral equation is cast as a system of simultaneous equations. The solution is a vector representing the (Laplace transform of the) radial pressure gradient evaluated at the wellbore.

Having obtained the Laplace transform of the radial pressure gradient, eq. (2-35) can be used to calculate the Laplace transform of the entire pressure field on the domain $\{0 < z < L; r > r_o\}$. The Laplace transformed expressions are inverted to the time domain using a numerical Laplace transform inversion algorithm (Stehfest, 1970; Crump, 1976).

Figures 2.4 and 2.5 show the computed pressure gradient at times of 10 hours and 100 hours after a unit pressure increase is applied at the surface. For these and the following figures, the formation is assumed to have a porosity of 20% and a permeability of 10 Darcy. The static barometric pressure is assumed to be 8.5E04 Pa and the gas viscosity to be 2.0E-05. 100 Fourier coefficients were retained in the Green's function summation and the number of terms retained in the numerical Laplace inversion algorithm was taken to be 4. The casing length was taken to be 100 m, the hole depth 500 m, and the depth to the "no-flow"

Figure 2.4. Radial pressure gradient evaluated at the hole face over the uncased portion of the open hole. $t = 10$ hrs.

Figure 2.5. Radial pressure gradient evaluated at the hole face over the uncased portion of the open hole. $t = 100$ hrs.

boundary 1000 m. The hole radius was taken to be 0.1 m. The rank of the kernel matrix (i.e. the number of nodes on the z and z' axes) was taken to be 30. The computer code CAVITY, given in the Appendix, was used to obtain these and the following results.

As expected, the pressure gradient at the wellbore is positive in the upper part of the hole and negative in the lower part, reflecting gas influx and outflow, respectively. The pressure gradient is roughly anti-symmetric about the mid-point of the open hole section for late times. However, for early times, the gradient is highly skewed, with the gas entry being heavily weighted to the region immediately below the casing. For early times, the gas exit from the cavity is more uniformly distributed with depth. The latter reflects the fact that, for early times, the perturbation due the applied surface pressure has not reached the deeper regions of the formation, so the formation pressure at these depths is almost uniform. Thus, the mass flux from the isobaric cavity will in turn be almost uniform.

Figures 2.6 and 2.7 show the vertical pressure distribution along the wellbore face. Also shown is the pressure distribution obtained from the 1-D "no-hole" solution. The latter also has a no-flow condition at $z=L$. The high frequency oscillations result from the "Gibb's phenomenon" of retaining only a finite number of terms in the Fourier series. However, the basic structure of the pressure profile is evident. Inspection of Figs. 2.6 and 2.7 suggests that the cavity pressure at any time might be closely approximated by simply averaging the 1-D (vertical flow) solution over the open hole section $[L_c, L_h]$. To check the validity of this hypothesis, we averaged both the 2-D solution on this interval (to dampen the ringing) and the 1-D solution. Figure 2.8 shows the time history of both approximations for the case under consideration. It was found that the two solutions agreed to within a few percent. This observation suggests that, under the assumption of a homogeneous and isotropic formation, considerable computational effort can be eliminated by using a vertically averaged 1-D solution as opposed to a 2-D (vertical and radial) flow model.

A theoretical argument can also be made for the validity of the vertically averaged 1-D model in representing the cavity pressure. Averaging eq. (2-38) over the uncased open hole interval $[L_c, L_h]$, and using eq. (2-37) as the definition of the (Laplace transform) of the cavity pressure, one can easily obtain the following expression for the Laplace transform of the cavity pressure in terms of the vertically averaged 1-D flow model:

$$\hat{R}(s) = \frac{1}{L_h - L_c} \int_{L_c}^{L_h} \hat{p}_h(z, s) dz + \frac{1}{L_h - L_c} \int_{L_c}^{L_h} dz' \int_{L_c}^{L_h} dz C_N(z|z') \left. \frac{\partial p}{\partial r} \right|_{r=r_0} \quad (2-39)$$

Figure 2.6. Vertical pressure profile from the surface to the no-flow boundary at $z = 1000$ m. 1-D vertical (no-hole) solution is superimposed on 2-D semi-analytic solution. $t=10$ hrs.

Figure 2.7. Vertical pressure profile from the surface to the no-flow boundary at $z = 1000\text{m}$. 1-D vertical (no-hole) solution is superimposed on 2-D semi-analytic solution. $t=100$ hrs.

Figure 2.8. Cavity pressure (orange) history due to a uni. increase in pressure at the surface at $t=0$. Solid line is 2-D semi-analytic solution. Dashed line is 1-D (vertically averaged) solution.

or, in the time domain,

$$p(t) = \frac{1}{L_h - L_c} \int_{L_c}^{L_h} p_h(z, t) dz + \mathcal{L}^{-1} \left[\frac{1}{L_h - L_c} \int_{L_c}^{L_h} dz' \int_{L_c}^{L_h} dz_N \langle z | z' \rangle \frac{\partial p}{\partial r} \right]_{r=r_0}$$

(2-40)

Clearly, based on the result displayed in Fig. 2-8, the expression containing the product of the Neumann Green's function must be quite small.

2.4 THE NEED FOR A MORE GENERAL MODEL, AND MORE SOPHISTICATED METHODS FOR USING THE MODEL TO INTERPRET BAROMETRIC DATA

Cases of practical interest in geophysics rarely satisfy the mathematicians desire to keep things simple. The flow models presented above are neat and tidy, and clearly have their place in the process of understanding how the formation gas (air) pressure or the cavity pressure responds to a surface barometric load. If nothing else, they are clearly necessary to test the precision of more sophisticated models. However, the upper few kilometers of the Earth's crust are not homogeneous and isotropic. In particular, permeability will almost always vary with depth.

For the near surface regime (upper few kilometers), this variation will be governed primarily by lithologic differences with depth. Pore compaction, and subsequent reduction in permeability with depth, is probably a second order effect at these depths. The Nevada Test Site presents an added challenge. In addition to naturally occurring spatial permeability variations, some areas of the NTS have what may prove to be significant artificially induced permeability due to shock wave deformation and fracture during a shot and also to stress relief about a collapsing cavity after a shot. If this weren't enough to age the modeller, consider the permeability of the ash fall, bedded tuffs, and rhyolite flows in the mesa region of the NTS (eg. Pahute and Ranier Mesas) and underlying the alluvial fan areas. Cooling cracks following initial emplacement of these tuffs will introduce a highly anisotropic permeability, more than likely higher in the vertical direction than the horizontal. Rhyolite flows, on the other hand, will probably tend to introduce barriers to vertical flow due to the slow cooling of the interior of the flow, and allow for free horizontal flow at the vertical extremities of the flow introduced by rapid cooling cracks.

The degree of sophistication of mathematical or numerical models must be tempered with the amount and nature of available data for interpretation. Over-parameterization of models can lead the modeller down a path of no return unless great care is used. The corollary to this is: "any data set can be matched by a model if enough flexibility is embedded in the model". For the case of diffusion processes, over-parameterization can easily occur. This is due to simply to the nature of the diffusion equation, whose solutions can be cast as a series of exponentials. The inverse problem, particularly when the measured signal is contaminated with any amount of noise, is notoriously ill-posed. This is the primary reason why conventional well testing models that are used to interpret field data rarely contain more than a few free parameters (eg. permeability, storage, skin, fracture length, etc.). More sophisticated reservoir models can account for a greater degree of complexity of a reservoir, but this complexity must be constrained by actual field measurements. Based on these considerations, we suspect that the barometric pressure method, in which the hole intersects several distinct layers, can probably extract some information regarding each of the layers. The obvious candidate target, presently the least understood at the Nevada Test Site, is permeability.

The barometric pressure method has, in addition to the simplicity of data acquisition, another advantage over conventional well testing procedures. This advantage is the frequency content of the source. Conventional well testing procedures generally flow a well at constant flow rate or constant pressure. In the frequency domain, this source is weighted very strongly at the low frequencies, except at very early times (first few seconds of flow). As a result, some detail may be lost if certain features of the reservoir (eg. fracture storage) are excited only at high frequencies. A discussion of this effect is given elsewhere (Hanson, 1983). Indeed, it has been only during the past few years that rapid data acquisition immediately after initiation of flow or shut-in has been recognized to yield valuable information.

Barometric pressure signals have a rich spectrum in comparison. Weather fronts have a varied, but characteristically low, frequency content. The higher frequencies are introduced by the atmospheric tides.

Based on the above discussion, it is not unreasonable to assume that a knowledge of the vertical permeability structure surrounding a partially cased hole can be realized using the barometric pressure testing method. The degree of sophistication of the model may be limited to several horizontal layers which are in themselves homogeneous and isotropic. However, these assumptions are similar to those made in conventional well testing methods. It is assumed here that the locations and thicknesses of the layers has been obtained a priori from logs. If porosity information is also available, from logs or cores, further constraints can be placed on the interpretation.

There is a temptation at this juncture to suggest that, because the vertically averaged 1-D model does such a remarkable job at representing the cavity pressure for a hole intersecting a homogeneous and isotropic half-space, it may also be used for the case in which the hole intersects a layered half-space. In order to test this hypothesis, a model must be constructed for the latter. Considerable effort has been expended during the course of this investigation in (attempting to) approximate the more complex situation using an analytic and semi-analytic as above for the homogeneous formation situation.

Unfortunately, a computationally practical scheme for doing so was not found. Such a procedure requires the evaluation of a series solution of Bessel-type functions for each layer. The coefficients of these series must be adjusted appropriately so that the interface flux and continuity boundary conditions are met. In addition, a Fredholm integral equation must be solved numerically as in the case presented above. Therefore, it was decided to proceed with a totally numerical (finite difference) model, which is discussed in the next chapter.

CHAPTER III

A NUMERICAL MODEL FOR 2-D FLOW IN A LAYERED HALF-SPACE

The numerical model for gas flow in and around a finite cylindrical hole intersecting a layered half-space was constructed using a finite difference solution to the non-steady state, two-dimensional flow in a nonhomogeneous half-space given by:

$$\frac{\mu}{P_b} \frac{\partial p}{\partial t} = \frac{\partial}{\partial z} \left(\frac{k}{\phi} \frac{\partial p}{\partial z} \right) + \frac{1}{r} \frac{\partial}{\partial r} \left(\frac{kr}{\phi} \cdot \frac{\partial p}{\partial r} \right) \quad (3-1)$$

P_b is the mean gas pressure in the formation and p is the variation of pressure about this mean. Note that, in the above equation, vertical (or horizontal) variations in both permeability and porosity are allowed. In application, these variations will be constrained to be constant within each layer. Figure 2.2 shows the generalized geometry of this problem.

The numerical solution to eq. (3-1), subject to initial conditions and appropriate boundary conditions [see eqs. (2-27)], is based on a previously published computer code (Frickett and Lonnquist, 1971). This finite difference code uses an iterated alternating direction implicit (IADI) technique to solve the difference equations at each time step. Details of the IADI method, and its implementation in terms of a computer code, are given in the above reference as well as most discussions of finite difference techniques. The computer code allows for up to 10 discrete layers, each having an assigned permeability and porosity. At the intersection of each layer with the wellbore, several sub-layers can be defined, each having different well completion characteristics. These include: (1) a no-flow boundary condition (eg. casing), (2) an open hole, and (3) no hole. The code is set up to be user-interactive.

The finite difference mesh is equi-spaced in the vertical dimension within each layer (or sub-layer), and is geometrically spaced in the radial dimension. The radial mesh is fine near the wellbore and gets larger in a geometric progression further from the wellbore. This accommodates the very large pressure gradients near the wellbore. The number of vertical nodes in within each layer is supplied by the user, with a maximum number of vertical nodes for the half-space of 200.

For the forward problem, the in-hole pressure variation is

calculated based on the user-supplied half-space geometry (including layering, and permeability and porosity values) and a surface pressure vs. time data file. For the inverse problem, the user has a choice of several methods, listed in increasing computational efficiency: (1) Monte Carlo, (2) Grid Search and (3) Ridge Regression. The inverse solution is simply a nonlinear least squares method using some sort of iteration sequence. Each of the above-listed methods is an iteration technique, where the model parameters are modified in some prescribed way from one iteration to the next. A solution is obtained when the rms difference between the measured in-hole pressure record and the predicted in-hole pressure record is minimized. This rms difference, called the "chi-square", is a measure of the "goodness of fit" of the solution. Statistical methods, such as the chi-square test, allow one to place a measure of confidence on the model used to interpret the measured data. Figure 3.1 shows a generalized flow chart for the inverse problem.

The computer codes RUNBFD and BFD, listed in the Appendix, allow for calculating forward models. RUNBFD sets up the finite difference mesh and initializes the formation parameters. As currently set up, these codes can address only the problem of an applied surface step function increase in pressure. The codes were written to be user-interactive.

Computer code KOSRA, given in the Appendix, can in addition to the forward, evaluate the inverse solution using any of the techniques listed above. A chi-square hypothesis test is carried out automatically to test the models. For the case of the grid search and ridge regression methods, the uncertainty in the model parameters is also calculated. Subroutines published in Bevington, 1969, have been used heavily in the inversion iteration schemes and also in the statistics. The code has been validated by comparison with 1-D (vertical) semi-analytic flow models for multiple layers and by comparison with the semi-analytic 2-D (vertical and radial) flow model presented in Chapter 2 for the homogeneous and isotropic half-space. The code was developed for an IBM-PC with 640 K memory, an 8087 co-processor, and a hard disk data storage device.

A forward model run for a typical data set (350 hours, one-half hour sample interval) takes several hours depending upon the node density in the finite difference mesh. As a result, the inverse problem is much too slow to be practical for a microcomputer. Because of this, the code was transported to both the CRAY and CDC 7600 machines at LLNL to carry out the multi-layer inversions presented in Chapter 5. Indeed, even on these fast machines, the inverse problem takes considerable computation time. For the examples given in Chapter 5, computation time to achieve a solution ranged from one-half hour to several hours. As a result, very few multi-layer inverse models were run.

It should be pointed out here that, because this numerical solution can, with a small amount of modification, address the problem of lateral (horizontal) changes in permeability, it is

Figure 3.1. Computational flow chart for nonlinear least-squares data inversion.

ideally suited to nuclear chimney analysis. To first approximation, the latter can be represented by a highly rubblelized zone of roughly cylindrical geometry intersecting a layered half-space. The cylindrical region is likely to have an anisotropic permeability distribution (vertical and radial) due to the nature of nuclear cavity collapse. By monitoring air pressure at the surface and at one or several positions within the chimney (accessed by slant drill holes from the surface), the magnitude of gas permeability and degree of permeability anisotropy within the chimney can likely be estimated. The success of this application will require that the vertical permeability distribution in the undisturbed layered half-space be known a priori. The techniques presented in this report will allow for establishing this vertical permeability distribution.

CHAPTER IV

EXAMPLES OF FORWARD MODELS

In this chapter, several examples are presented of the 1-D and 2-D flow models presented in Chapters 2 and 3. The semi-analytic 1-D models have been constructed principally for the purpose of validating the more sophisticated 2-D numerical (finite-difference) flow model discussed in the previous chapter. Computer codes used to calculate these examples are given in the Appendix. We have taken three examples to demonstrate the flexibility in the software and to verify the usefulness of the 1-D (vertically averaged) flow model for interpreting barometric pressure transient data under the assumption of a uniform (homogeneous and isotropic) half-space.

The first example considered is that of a homogeneous and isotropic half-space. A hole of radius 0.1 m and depth of 500 m is cased to 100 m in a formation of 20% porosity and 10 Darcy permeability. For the finite difference calculations, no-flow boundaries were placed at a depth $z=1000$ m and radial distance $r=1000$ m. Figure 4.1 shows the geometry. The initial gas pressure in the formation is 85 kPa. At $t = 0$, a pressure increase of 1000 Pa is applied at the surface and maintained. Figure 4.2 shows the in-hole pressure change as a function of time. Superimposed on this graph is the 1-D (vertically averaged) solution, which averages the "no-hole" solution over the uncased part of the open hole. Computer code BFD (see Appendix) was used to calculate the 2-D solution, and code HOMO03 (also see Appendix) was used to calculate the 1-D (vertically averaged) solution. It is evident that the less sophisticated 1-D (vertically averaged) solution is an excellent approximation to the true solution for the case of a homogeneous half-space.

Figure 4.3 shows the radial pressure gradient evaluated at the wellface as calculated by code BFD at $t=100$ hrs. This figure demonstrates the nature of gas flow into and out of the hole. Just beneath the casing, the pressure gradient is highest, indicating the highest flow rate into the hole. Progressing down the hole, we find a position at which there is no flow into or out of the hole. In the lower part of the hole, the gas flow is out of the hole. Figure 4.4 shows the pressure profile with depth along the axis of the hole and at the outer radial boundary of the finite difference mesh. Superimposed on the latter is the vertical pressure profile evaluated by the 1-D solution (code HOMO02) as a check on the finite difference calculation.

The second example, shown in Figure 4.5, is slightly more complex than the first. Figure 4.6 shows the time history of the pressure change from initial conditions due to the application of a 1000 Pa pressure change at the surface at $t=0$. Superimposed on this history is the 1-D (vertically averaged) solution. It is evident from this figure that, if the formation deviates from the

MODEL 'A'

Figure 4.1. "Model A". Partially cased hole intersecting a homogeneous and isotropic half-space.

Figure 4.2. Computed cavity pressure change vs. time for the finite difference 2-D flow model (solid line) and 1-D (vertically averaged) model (dashed line) - Model A.

WELLFACE PRESSURE GRADIENT AFTER 100 HOURS ... MODEL A

Figure 4.3. Radial pressure gradient evaluated at the wellface using the finite-difference 2-D flow model after 100 hrs.- Model A.

Figure 4.4. Pressure change along the cavity axis (dashed line) and at the outer radial no-flow boundary (solid line) computed using the finite-difference 2-D flow model after 100 hrs. Superimposed on the outer boundary profile is the semi-analytic 1-D solution as a check.

MODEL "B"

Figure 4.5. "Model B". Partially cased hole intersecting a layered half-space.

Figure 4.6. Computed cavity pressure change vs. time for the finite difference 2-D model (solid line) and the 1-D (vertically averaged) model (dashed line) - Model B.

homogeneous and isotropic situation, the 1-D vertically averaged solution becomes a poorer approximation. Figure 4.7 shows the radial pressure gradient evaluated at the wellface. Figure 4.8 shows the vertical pressure profile calculated along the axis of the wellbore and the same evaluated at the outer radial boundary. Superimposed on the latter is the 1-D solution calculated by the semi-analytic solution code HOMOG2.

The third example, which is the most complex, is shown in Figure 4.9. This 6-layer model has widely varying layer thicknesses, porosities, and permeabilities over the open-hole section. Figure 4.10 shows the time history of the change of the in-hole pressure from initial conditions due to an increase in surface pressure of 1000 Pa at $t=0$. Superimposed on this history is that calculated by the 1-D (vertically averaged) solution over the uncased section of open hole. It is very clear that the latter is a very poor approximation to the true prediction. Figure 4.11 shows the radial pressure gradient evaluated at the wellface. Figure 4.12 shows the vertical pressure profile along the axis of the hole and also at the outer radial boundary. Superimposed on the latter is the 1-D flow model vertical pressure profile calculated by code HOMOG2.

Figures 4.13 and 4.14 show the wellface pressure gradient and axis pressures of the three examples superimposed on the same plot. It is evident from these figures that a modification of the formation from a simple homogeneous and isotropic formation to a layered half-space can alter the flow field and in-hole pressure history in a quite significant way. This suggests that, if the vertical variations in either permeability or porosity are sufficiently large, an appropriate analysis of the barometric response of the cavity to surface pressure fluctuations may yield information concerning the vertical distribution of flow properties in the formation.

Figure 4.7. Radial pressure gradient evaluated at the wellface using the finite-difference 2-D flow model after 100 hrs. - Model B.

PRESSURE VS DEPTH AT WELL AXIS AND OUTER BOUNDARY AFTER 100 HOURS...
MODEL B

Figure 4.8. Pressure change along cavity axis (dashed line) and along the outer radial no-flow boundary (solid line) computed using the finite-difference 2-D flow model. Superimposed on the outer boundary profile is the semi-analytic 1-D solution as a check.

MODEL "C"

Figure 4.5. "Model C". Partially cased hole intersecting a layered half-space.

Figure 4.10. Computed cavity pressure change vs. time for the finite difference 2-D model (solid line) and the 1-D (vertically averaged) model (dashed line) - Model C.

Figure 4.11. Radial pressure gradient evaluated at the wellface using the finite-difference 2-D flow model after 100 hrs. - Model C.

PRESSURE VS DEPTH AT WELL AXIS AND OUTER BOUNDARY AFTER 100 HOURS...
MODEL C

Figure 4.12. Pressure change along cavity axis (dashed line) and along outer radial no-flow boundary (solid line) computed using the finite-difference 2-D flow model. Superimposed on the outer boundary profile is the semi-analytic 1-D solution as a check.

WELLFACE RADIAL PRESSURE GRADIENT FOR MODELS A,B,AND C

Figure 4.13. Radial pressure gradients evaluated at the wellface after 100 hrs. for Models A, B, and C.

AXIS PRESSURE FOR MODELS A,B,AND C

Figure 4.14. Change in cavity axis pressures after 100 hrs. for Models A, B, and C.

CHAPTER V

ANALYSIS OF BAROMETRIC PRESSURE DATA FROM U2fe U19ac, AND U20ai AT THE NEVADA TEST SITE

Barometric pressure data taken at holes U2fe, U19ac, and U20ai were analyzed in terms of the flow models discussed previously. Figure 5.1 shows the Area designation at the Nevada Test Site. Hole U2fe is located in Area 2, hole U19ac is located in Area 19, and U20ai is located in Area 20. More detailed location maps for these two holes will be given as the analysis of each of the holes is discussed.

Data acquisition consisted of monitoring surface and in-hole pressure over periods of time ranging roughly from 250 hours to 800 hours. In-hole instrumentation consisted of a high precision quartz pressure gauge suspended roughly 20 feet from the top of the casing. The hole was isolated from direct hydraulic communication to the surface. This seal consisted of a steel plate welded to the top of the casing. Electrical connections to the in-hole pressure gauge were made through an air-tight electrical feed-through. Surface barometric pressure was monitored with two identical quartz pressure gauges. Both in-hole and surface pressure gauges were interrogated simultaneously every half-hour and pressure data recorded using a Digiquartz Data Logger Model 750. The quartz pressure gauges used were Digiquartz Pressure Transducer Model 215-AS-012 (0-15 psi) with a calibration accuracy of 0.015% of full scale. The quartz sensors and the data logger are products of Paroscientific, Inc. Surface sensors and the data logger were housed in an air-conditioned instrumentation trailer near the wellhead.

5.1 Analysis of Data Taken at NTS hole U2fe

Figure 5.2 shows the pre-shot location of emplacement hole U2fe relative to previous shots in the area. This hole was spudded on September 1, 1979 and completed to a depth of 450 m. on January 11, 1980. The drill bit size was 2.438 m diameter. A casing was set to 32.6 m depth and cemented. Figure 5.3 shows the geologic cross section through U2fe. Table 5.1 describes the lithology for various depth intervals.

Barometric pressure data was monitored between May 18, 1982 and June 2, 1982. Figures 5.4, 5.5 and 5.6 show the recorded pressure data for the two surface sensors and the down hole sensor, respectively. Figures 5.7, 5.8 and 5.9 show the Fourier amplitude spectra of these data sets, respectively. The arithmetic mean of each of the data sets was subtracted from the signal prior to calculating the spectra. The frequency domain plots show significant energy at the tidal frequencies. These peaks correspond with the diurnal (24 hour period), semidiurnal (12 hour period), and tidal (8 hour period) tidal energy bands. A more detailed discussion of these tidal components will

Figure S.1. Map showing Area locations at the Nevada Test Site.

Figure 5.2. Pre-shot location map of emplacement hole U2fe. (from Howard, 1982).

Figure 5.3. Geologic cross-section through emplacement hole U2fe. (from Howard, 1982).

TABLE 5.1

DESCRIPTION OF LITHOLOGY AT 02fe
(from Howard, 1982)

<u>DEPTH (m)</u>	<u>DESCRIPTION</u>
	Alluvium 0 - 449.6 T.D.
0 - 335.3	Mixed alluvium, brown to gray-brown variable proportions of Paleozoic clastic and carbonate detritus and Tertiary tuff fragments, Paleozoic clasts are more abundant than tuff clasts, calcaceous sand-silt matrix, caliche rinds on clasts are common, the alluvium consists of alternating beds of sand and gravel, with locally very coarse boulder beds, CO ₂ ranges from 0.2 - 13.7%, and averages 6.0%.
335.3 - 378.0	Mixed alluvium, brown to gray-brown, tuff detritus comprises >75% of the sediment, and Paleozoic clasts <25%, alternating beds of sand and gravel, with local coarse boulder beds, caliche is less common than unit above, CO ₂ ranges from 0.3 - 3.3%, and averages 2.6%.
378.0 - 449.6 T.D.	Mixed alluvium, brown to light brown, highly tuffaceous, tuff detritus comprises 80-90% and Paleozoic clasts, 10-20% of the sediment, basalt fragments occur at 393 m, the sediment ranges from mostly gravelly sand at the top of unit to coarse gravels at the base, caliche is minor, CO ₂ ranges from 0.1 - 2.9%, and averages 0.4%.
GLH 10/11/82	Lithology is based on drill cuttings, sidewall samples, grain density and CO ₂ analyses, and gamma log and downhole movie interpretation.

Figure 5.4. Surface sensor #1 pressure record - U2fe.

Figure 5.5. Surface sensor #2 pressure record - U2fe.

Figure 5.6. In-hole sensor pressure record - Uéfe.

Figure 5.7. Amplitude spectrum of surface sensor #1 - U2fe.

Figure 5.8. Amplitude spectrum of surface sensor #2 - U2fe.

Figure 5.9. Amplitude spectrum of in-hole sensor - U2fe.

Figure 5.10. "Best-fit" of predicted in-hole pressure (dotted) using surface sensor #1 data and 1-D (vertically averaged) model, superimposed on measured in-hole pressure data (solid) - U2Fe.

Figure 5.11. "Best-fit" of predicted in-hole pressure (solid) using surface sensor #2 data and 1-D (vertically averaged) model, superimposed on measured in-hole pressure data (dotted) - U2fe.

be given later in this report. It is clear from both the time domain and frequency domain plots that the in-hole pressure signal amplitude is damped relative to the measured surface pressure signal. The degree of damping increases with increasing frequency as expected.

An initial estimate of the formation hydraulic diffusivity at U2fe was obtained by assuming that the formation is homogeneous and isotropic. With this assumption, the 1-D flow model (averaged over the open hole section between the bottom of the casing and the bottom of the hole) is applicable. As previously shown, the 1-D vertically averaged model is an excellent approximation to the 2-D (vertical and radial) flow model, and is computationally much more efficient.

Data analysis consisted of using the grid search method and varying the one free parameter (hydraulic diffusivity) in the model. Analysis was carried out in the time domain. In the time domain, the pressure response in the hole at any given time depends on the history of the surface pressure record up to and including that time. At the beginning of the pressure record, we clearly do not have that history available to us. Therefore, the predicted pressure response will not be valid until some time after the start of the recorded pressure record. We chose 100 hours as a conservative upper bound on this "formation equilibration time" based on (a) the casing depth and (b) typical permeability and porosity values for unconsolidated sandstones. In the nonlinear inversion, the rms difference between the predicted in-hole pressure and measured in-hole pressure is therefore minimized for the section of data corresponding with 100 hours to the end of the record. Arithmetic means of the pressure records were subtracted prior to analysis to eliminate the pressure difference due to the elevation difference between the surface sensors and the in-hole sensor.

Figure 5.10 shows the "best fit" of the predicted in-hole pressure with the measured in-hole pressure based on the above assumptions for the surface pressure data recorded by sensor #1. Figure 5.11 shows the equivalent for surface pressure data recorded by sensor #2. Table 5.2 shows the calculated parameters corresponding to these two data inversions.

TABLE 5.2

COMPUTED DIFFUSIVITIES AT NTS HOLE U2fe

Surface Sensor	Diffusivity (m ² /hr)	Uncertainty (m ² /hr)	Reduced Chi-Square
#1	4806	117	33
#2	4726	190	23

The uncertainty indicated is at the 1-sigma level. Also, the pressure measurement uncertainty was assumed to be 5 Pa (0.000725 psi).

A porosity of 9.3% was assigned to the half-space based on a weighted average of measured porosities (from hole logs) where the weighting factor was proportional to the section thickness. Using this value of porosity, and values of viscosity and mean formation pressure of 2.0e-05 Pa-sec and 8.5e05 Pa, respectively, the following values of permeability were calculated:

TABLE 5.3

COMPUTED PERMEABILITIES AT NTS HOLE U2fe

Surface Sensor	Permeability (Darcies)	Uncertainty (Darcies)
#1	29.6	0.7
#2	29.1	1.2

In order that the model used be considered a "sufficient model" in terms of adequately fitting the data under the constraint of a known measurement uncertainty, the reduced chi-square should be approximately unity. The above values are significantly higher than this value. This could be a consequence of several factors: (1) sensor-specific problems including: the pressure measurement uncertainty is too small, poor sensor calibration, long term drift, or lack of correcting for temperature effects on the sensor or (2) model-specific problems including: the model is too simple and does not represent the true situation.

Sensor-specific problems are difficult to address without a

significant laboratory environment testing effort. However, there is a reasonable possibility that the measurement uncertainty chosen was too small. Winds on the surface could introduce a high frequency noise component on the surface sensors. Also, the air-conditioning system in the instrumentation trailer could introduce a noise-like perturbation to the surface sensors as it switches on and off. We should point out here that, if the measurement uncertainty were increased by a factor of 2, the reduced chi-square would decrease by a factor of 4. Thus, the "validity" of the model is very sensitive to the level of measurement uncertainty (either device dependent or to natural background noise).

Analysis of the U2fe data was carried out using more complex flow models than that described above. These included 2- and 3-layer models. Since it has been shown previously that the vertically averaged 1-D (vertical) flow model is not applicable to these more complex situations, the full 2-D finite difference flow model was used. A single layer model was also evaluated using the 2-D flow finite-difference model to confirm the above results which are based on the 1-D vertically averaged model. In the 2-D finite-difference model, each layer has assigned a value of porosity and permeability. Porosity in each layer was constrained by values measured from hole logs. Both grid search and ridge regression non-linear inversion techniques were used out to estimate permeability in each layer. No-flow boundaries were placed at $z=1000$ m and $r=1000$ m to simulate conditions at infinity. The results are displayed diagrammatically in Figure 5.12.

The multilayer models illustrated in Figure 5.12 clearly show a tendency for permeability to increase with depth, at least to a depth of 334 m. Lithology logs suggest that gravels get somewhat coarser with depth and that caliche is more prevalent near the surface. These observations are consistent with the increasing permeability with depth. Note also that, as the model sophistication increases, the reduced chi-square decreases, indicating that these may be more appropriate models than the simple uniform half-space.

It is unclear why the inversion of the U2fe data using the assumption of a homogeneous and isotropic half-space yielded different answers for the 1-D (vertically averaged) solution and the full 2-D model. The former yielded roughly 29 Darcies and the latter yielded 47 Darcies. One possible reason for this is that the 1-D vertically averaged solution assumes a true half-space, whereas the 2-D numerical model assumed a no-flow boundary at $z=1000$ m. The characteristic diffusion length for diffusivities on the order of 5000 m^2/hr and signal wavelengths of 100 hrs is roughly 700 m. This is getting close to the 1000 m no-flow boundary, which may have been set too shallow. If this were the case, then the calculated permeability may indeed be high for the assumption of a true half-space geometry.

Taking into account, however, that the static water table at

Figure 5.12. Data inversion results on one, two, and three layer models using the finite-difference 2-D flow model - U2fe. Note that the three-layer model was terminated after one iteration, and the minimum in chi-square was not yet reached.

U2fe is at a depth of roughly 580 m, which represents an effective no-flow boundary, we prefer the higher value of permeability (47.1 Darcies).

5.2 Analysis of Data Taken at NTS hole U19ac

Figure 5.13 shows the drill hole location map for the Pahute Mesa and indicates the location of U19ac relative to boundaries of the Timber Mountain and Silent Canyon Calderas and to other holes in the vicinity. Emplacement hole U19ac was spudded on January 2, 1980 and drilled to a depth of 701 m by February 9, 1980. Two zones in the hole, 26.5-178.6 m and 353.6-426.1 m, lost circulation material during drilling and were subsequently stabilized with bentonite mud. Drill bit size was 2.44 m diameter. A 17.7 m deep casing ("surface conductor") was set and cemented prior to drilling.

Figure 5.14 shows the geologic cross section through U19ac. Geologic descriptions of the formations intersected by U19ac are given in Table 5.4. Figure 5.15 shows the NE-SW stratigraphic correlation with two nearby holes.

A total of five barometric pressure data sets were taken at U19ac over the period of September 16, 1982 and December 20, 1982. Annotation of the data records was sparse or non-existent regarding start and stop dates for each of the records. However, we have been able to reconstruct this information using continuity between records and record length. The following Table shows this reconstruction.

TABLE 5.5

START AND STOP TIMES FOR U19ac DAT SETS

PILE	START DATE	STOP DATE
01	9 /16/82	9/30/82
02	10/12/82	10/24/82
03	10/24/82	11/ 3/82
04	11/ 3/82	11/15/82
05	11/15/82	12/20/82

Data acquisition was identical to the procedure used for U2fe. Figures 5.16 through 5.30 show the recorded surface and in-hole pressure records for each of the above data sets. Note that the mean recorded pressure is less than that recorded at U2fe. This is due to the elevation difference between the two sites.

Figure 5.13. Pahute Mesa drill hole location map, showing location (see arrow) of hole U19ac. (from Howard, 1984).

Figure 5.14. Geologic cross section through U19ac. (from Howard, 1984).

TABLE 5.4
DESCRIPTION OF LITHOLOGY AT U19ac
(from Howard, 1984)

Depth (m)

TIMBER MOUNTAIN FORMATION

Ammonia Tanks Member

0-65 Ash-flow tuff, moderately to densely welded, generally devitrified with basal 3.0 meter vitric zone.

65-71 Tuffaceous sandstone and reworked bedded tuff, vitric.

Rainier Mesa Member

71-79 Ash-flow tuff, vitric, non- to partially welded.

79-101 Ash-flow tuff, moderately welded, vapor phase in part.

101-105 Vitrophyre

105-377 Ash-flow tuff, moderately to densely welded, generally devitrified.

377-383 Vitrophyre

383-395 Ash-flow tuff, partially to moderately welded, vitric.

PAINTBRUSH TUFF (395-532)

395-431 Ash-fall and reworked bedded tuffs, slightly calcareous from 408 to 431 m, devitrified and slightly argillized from 409-426 m.

431-440 Basalt Flow
Vesicular, brecciated, slightly calcareous.

440-532 Ash-fall and reworked bedded tuffs, vitric to partially zeolitized, erratic sparse blebs of secondary calcite.

PRE-PAM CANYON RHYOLITE LAVA (532-701 T.D.)

532-541 Lava flow, frothy breccia, vitric.

541-555 Lava flow vitrophyre, perlitic in part, vitric.

555-637 Lava flow breccia and non-brecciated banded lava flow, devitrified to vitric, zeolitized in part.

637-701 Lava flow breccia, devitrified, silicified in part, basal 7.0 meters (T.D.) slightly zeolitized.

DEPTH INTERVAL OF LITHOLOGIC UNIT

STRATIGRAPHIC UNIT

NE-SW Stratigraphic Correlation Diagram B-B'
(See surface geology map Fig. 9 for location of holes)

Figure 5.15. NE-SW stratigraphic correlation at U19ac. (from Howard, 1984).

Figure 5.16. Sur ice sensor #1 pressure record - File #1. U19ac.

Figure 0.17. Surface sensor #2 pressure record - File #1. U19ac.

Figure 5.18. In-hole sensor pressure record - File #... U19ac.

Figure 5.19. Surface sensor #1 pressure record - File #2. U19ac.

Figure 5.20. Surface sensor #2 pressure record - File #2, U19ac.

Figure 5.21. In-hole sensor pressure record - File #2, U19ac.

Figure 5.22. Surface sensor #1 pressure record - File #3, U19ac.

Figure 5.23. Surface sensor #2 pressure record - File #3. U19ac.

Figure 5.24. In-hole sensor pressure record - File #3, U19ac.

Figure 5.25. Surface sensor #1 pressure record - File #4. U19ac.

Figure 5.26. Surface sensor #2 pressure record - File #4. U19ac.

Figure 5.27. In-hole sensor pressure record - File #4. U19ac.

Figure 5.28. Surface sensor #1 pressure record - File #5, U19ac.

Figure 5.29. Surface sensor #2 pressure record - File #5. U19ac.

Figure 5.30. In-hole sensor pressure record - File #5, U19ac.

Figures 5.31 through 5.45 show the Fourier amplitude spectra of the pressure records for each of the above data sets.

As in the analysis of the U2fe data, an initial estimate of the formation hydraulic diffusivity was obtained by assuming that the formation intersected by U19ac is homogeneous and isotropic. With this assumption, the 1-D flow model (averaged over the open hole section between the bottom of the casing and the bottom of the hole) is applicable. Figures 5.46 through 5.55 show the "best fit" of the predicted in-hole pressure with the measured in-hole pressure based on the 1-D (vertically averaged) flow model. Table 5.6 shows the results of the grid search data inversion using this simple model.

TABLE 5.6

COMPUTED DIFFUSIVITIES FOR NTS HOLE U19ac

File No.	Surface Sensor	Diffusivity (mms ² /hr)	Uncertainty (mms ² /hr)	Reduced Chi-Square
01	#1	91640	4134	89
	#2	82280	2451	41
02	#1	51280	3120	64
	#2	66850	3160	34
03	#1	58700	2035	39
	#2	65820	1989	26
04	#1	50530	1623	79
	#2	56210	1495	48
05	#1	22710	675	40
	#2	23330	681	38

Two things are immediately apparent in the above Table. The first is that the magnitude of hydraulic diffusivity at U19ac is roughly an order of magnitude greater than that at U2fe. The second is that the hydraulic diffusivity decreases by almost a factor of five between September 16 and December 20.

The large magnitude of diffusivity can be attributed to the nature of the formations intersected by U19ac. These formations consist principally of bedded, ashfall, and ashflow tuffs. These volcanics, upon cooling, can shrink to create pervasive cracks that, depending upon cooling rate, be quite significant. As a result, the gas diffusivity can be large. A more complete discussion of these factors is given in a later chapter.

Figure 5.31. Amplitude spectrum, surface sensor #1, File #1, U19ac.

Figure 5.32. Amplitude spectrum, surface sensor #2. File #1, U19ac.

Figure 5.33. Amplitude spectrum, in-hole sensor, File #1, U19ac.

Figure 5.34. Amplitude spectrum, surface sensor #1. File #2, U19ac.

Figure 5.35. Amplitude spectrum, surface sensor #2, File #2, U19ac.

Figure 5.36. Amplitude spectrum, in-hole sensor, File #2, U19ac.

Figure 5.37. Amplitude spectrum, surface sensor #1, File #3, U19ac.

Figure 5.38. Amplitude spectrum, surface sensor #2, File #3.
U19ac.

Figure 5.39. Amplitude spectrum, in-hole sensor, File #3, U19ac.

Figure 5.40. Amplitude spectrum surface sensor #1, File #4, U19ac.

Figure 5.41. Amplitude spectrum, surface sensor #2, File #4, U19ac.

Figure 5.42. Amplitude spectrum, in-hole sensor, File #4, U19a2.

Figure 5.43. Amplitude spectrum, surface sensor #1. File #5. U19ac.

Figure 5.44. Amplitude spectrum, surface sensor #2, File #5, U19ac.

Figure S.45. Amplitude spectrum, in-hole sensor, File #5, U19ac.

Figure 5.46. "Best-fit" of predicted in-hole pressure (dotted) using surface sensor #1, File #1, data and 1-D (vertically averaged) model, superimposed on measured in-hole pressure data (solid) - U19ac.

Figure 5.47. "Best-fit" of predicted in-hole pressure (dotted) using surface sensor #2, File #1, data and 1-D (vertically averaged) model, superimposed on measured in-hole pressure data (solid) -U19ac.

Figure 5.48. "Best-fit" of predicted in-hole pressure (dotted) using surface sensor #1, File #2, data and 1-D (vertically averaged) model. Superimposed on measured in-hole pressure data (solid) - u13ac.

Figure 5.49. "Best-fit" of predicted in-hole pressure (dotted) using surface sensor #2, File #2, data and 1-D (vertically averaged) model, superimposed on measured in-hole pressure data (solid) - U19ac.

Figure 5.51. "Best-fit" of predicted in-hole pressure (dotted) using surface sensor #2, File #3, data and 1-D (vertically averaged) model, superimposed on measured in-hole pressure data (solid) - U19ac.

Figure 5.52. "Best-fit" of predicted in-hole pressure (dotted) using surface sensor #1, File #4, data and 1-D (vertically averaged) model, superimposed on measured in-hole pressure data (solid) - U19ac.

Figure 5.53. "Best-fit" of predicted in-hole pressure (dotted) using surface sensor #2, File #4, data and 1-D (vertically averaged) model, superimposed on measured in-hole pressure data (solid) - U19ac.

Figure 5.54. "Best-fit" of predicted in-hole pressure (dotted) using surface sensor #1, File #5, data and 1-D (vertically averaged) model, superimposed on measured in-hole pressure data (solid) - U19ac.

Figure 5.55. "Best-fit" of predicted in-hole pressure (dotted) using surface sensor #2, File #5, data and 1-D (vertically averaged) model, superimposed on measured in-hole pressure data (solid) - U19ac.

The decrease in hydraulic diffusivity with time can be attributed to precipitation during the time span over which the data was taken. The data listed in Table 5.7 was obtained courtesy of Dr. Joseph Hearst (LLNL, personal communication).

TABLE 5.7

PRECIPITATION RECORD AT HOLE U19ac

Time Span	Precipitation
9/24/82 - 9/30/82	2" rain
10/24/82	
10/30/82	0.65" rain
10/31/82	
11/9/82	0.5" rain
11/10/82	3 or 4" snow on Pahute Mesa

The decrease in hydraulic diffusivity can be a consequence of either (1) the near surface becoming water-saturated resulting in the connected pore space in the near-surface region becoming blocked creating an effective "skin effect" or (2) to a blanket of snow serving as a flow-barrier ("skin") to the more permeable formations below. One should also note that the uncertainty in the calculated hydraulic diffusivity becomes smaller as the diffusivity becomes smaller. This is due to the fact that, with a reduction in diffusivity, there is a greater difference between the system input (measured surface pressure) and system output (measured in-hole pressure), which results in a steeper depression near the minimum of chi-square. This in turn results in a better estimate of diffusivity.

In order to translate hydraulic diffusivity to permeability, an estimate of porosity is required. Evaluation of connected porosity is less straight forward in a fractured formation than in a simple porous sandstone. This is due to scaling considerations. Depending on fracture spacing, connectivity, and aperture, different values of connected porosity can be obtained depending on the averaging volume taken. Hole logs should be considered somewhat suspect since only a relatively small volume is sampled. Also, they cannot differentiate between connected and closed porosity. Since connected porosity is so difficult to estimate, we calculated permeability based on two assumptions: (a) porosity of 0.1 and (b) porosity of 0.01. One should keep in mind that connected porosity in fractured hard rock can be much less than 0.01 (Snow, 1968). Since permeability is directly proportional to porosity, the values of permeability presented

below can be scaled accordingly.

TABLE 5.8
COMPUTED PERMEABILITIES AT NTS HOLE U19ac

File No.	Surface Sensor	Porosity	Permeability (Darcies)	Uncertainty (Darcies)
#1	#1	0.1	599	27
	#2	0.1	538	16
	#1	0.01	59.9	2.7
	#2	0.01	53.8	1.6
#2	#1	0.1	335	20
	#2	0.1	437	21
	#1	0.01	33.5	2.0
	#2	0.01	43.5	2.1
#3	#1	0.1	384	13
	#2	0.1	430	13
	#1	0.01	38.4	1.3
	#2	0.01	43.4	1.3
#4	#1	0.1	330	11
	#2	0.1	367	10
	#1	0.01	33.0	1.1
	#2	0.01	36.7	1.0
#5	#1	0.1	148	4.4
	#2	0.1	152	4.5
	#1	0.01	14.8	0.4
	#2	0.01	15.2	0.5

One must keep in mind that it is hydraulic diffusivity and not permeability that governs mass transport. For the case of the Nevada Test Site, where we are primarily concerned with transport of radioactive gas to the surface, diffusivity and not permeability is the key parameter.

5.3 Barometric Tide Contamination Effects

The solid Earth tidal spectrum consists primarily of five tidal constituents: the O1 and K1 tides in the diurnal energy band and the M2, M2, and S2 tides in the semi-diurnal energy band. Of these, the K1 and S2 tides are of solar origin, the remainder being of lunar origin. In addition to these frequencies, the barometric tidal spectrum is also effected by the solar tides S1 and S3. Because the K1, S1, S2, and S3 tides are solar, there is a distinct possibility that at these frequencies, there may exist a signal contamination due to thermal effects. Every solar day, the surface conductor and sealing plate is subject to solar radiation which will result in a slight increase in the air temperature within the hole. The heat energy is transferred to the gas via conduction through the top of the casing and the surface seal. The energy may be distributed throughout the gas column within the hole due to convection.

In any event, the slight rise in temperature may result in a small rise in pressure within the hole which is independent of the surface barometric pressure. We point out here that, as in oceanic tides, the atmosphere also rises and falls due to gravitational effects, although the gravity effect is much smaller than thermal effects. Wellhead heating, however, will introduce a contamination that is amplitude and phase shifted from the normal surface barometric pressure at these frequencies.

In order to check the degree of thermal contamination in the measured in-hole pressure record, we carried out a line spectrum analysis at the tidal frequencies (Godin, 1972; Hanson, 1984) for the above-listed tides on both the measured and predicted in-hole pressure record for file #5 at U19ac. The predicted in-hole pressure is based on the best fitting 1-D (vertically averaged) flow model. The fact that this is an exceedingly over-simplified model does not affect the conclusions regarding thermal contamination. This is due to the fact that both solar and lunar tides are very close in frequency for the respective energy bands in the tidal spectrum. Thus, if there were no thermal contamination due to wellhead heating, adjacent (in the frequency domain) solar and lunar tides would show very similar amplitude and phase shifts with respect to the surface signal at these frequencies. Solar contamination will be manifested in adjacent tidal constituents (solar and lunar) of the measured in-hole pressure record showing substantially different amplitudes and phases from the predicted tidal constituents.

A zero-phase digital low pass filter was applied to both the measured in-hole and predicted in-hole pressure data for file #5 of the U19ac data. The filter in the time domain is shown in Figure 5.56 and in the frequency domain in Figure 5.57. The low passed data was then subtracted from the original data set to obtain a resultant signal corresponding primarily to the tidal constituents plus any existing higher frequency signal plus noise. These data were line spectral analyzed assuming that the

Figure 5.56. Zero-phase low-pass filter in the time domain.

Figure 5.57. Zero-phase low-pass filter in the frequency domain.

data contained, in addition to noise, the following tidal frequency set:

TABLE 5.9

ATMOSPHERIC TIDAL CONSTITUENTS

Tidal Constituent	Frequency (cph)
O1	0.0387306544
S1	0.0416667210
K1	0.0417807462
N2	0.0789992487
M2	0.0805114006
S2	0.0833333333
S3	0.1250000000

Figure 5.58 shows the tidal spectrum of the measured (filtered) surface data in terms of 90% confidence interval ellipses. The ellipse nature is only evident when these data are plotted on the real and imaginary axes. When plotted on amplitude-phase axes the ellipses become distorted. Figures 5.59 and 5.60 show the spectra of the measured and predicted in-hole pressure data, respectively. The confidence intervals assume that all signal not corresponding to the tidal constituents is random noise. The K1 and M2 tidal constituents were not resolved in the predicted in-hole pressure record. By not resolved, we mean that the 90% confidence ellipse intersected the origin of the amplitude-phase plot. The lack of resolution of the M2 tide in the predicted in-hole pressure record is due to the fact that the driving signal (surface pressure) indicates a very small amplitude at this frequency. Note also that there is a an amplitude reduction and phase shift between the measured surface pressure and predicted in-hole pressure for all of the tidal constituents. This is a consequence of the flow model.

It is clear from these spectra that there is significant contamination of the in-hole pressure record, particularly at the principle solar tides S1, K1, and S2. Modelling the thermal effect of the heated surface conductor would be very difficult. Therefore, we low-passed all of the data (surface and in-hole) for U2fe and U19ac to simply eliminate all of the information, contaminated or otherwise, at the tidal frequencies.

Figures 5.61 through 5.72 show the (filtered) measured in-hole data superimposed on the best fitting predicted in-hole pressure data using the 1-D (vertically averaged) model. The surface pressure data used to obtain the prediction was filtered also. The same digital low-pass filter was used in all filtering operations. Table 5.10 shows the recomputed hydraulic

Figure 5.58. 90% confidence envelopes of principle tidal constituents, surface sensor #1, File #5 - U19ac.

Figure 5.59. 90% confidence envelopes of principle tidal constituents, measured in-hole pressure, File #5 - U19ac.

Figure 5.60. 90% confidence envelopes of principle tidal constituents, predicted in-hole pressure, File #5 - U19ac.

Figure 5.61. "Best-fit" of predicted in-hole pressure (dotted) using filtered surface sensor #1 data and 1-D (vertically averaged) model. Superimposed on measured, filtered in-hole pressure data (solid) - U2fe.

Figure 5.62. "Best-fit" of predicted in-hole pressure (dotted) using filtered surface sensor #2 data and 1-D (vertically averaged) model, superimposed on measured, filtered in-hole pressure data (solid) - U2Fe.

Figure 5.63. "Best-fit" of predicted in-hole pressure (dotted) using filtered surface sensor #1, File #1 data and 1-D (vertically averaged) model, superimposed on measured, filtered in-hole pressure data (solid) - U19ac.

Figure 5.64. "Best-fit" of predicted in-hole pressure (dotted) using filtered surface sensor #2, File #1 data and 1-D (vertically averaged) model, superimposed on measured, filtered in-hole pressure data (solid) - U19ac.

Figure 5.65. "Best-fit" of predicted in-hole pressure (dotted) using filtered surface sensor #1, File #2 data and 1-D (vertically averaged) model, superimposed on measured, filtered in-hole pressure data (solid) - U19ac.

Figure 5.66. "Best-fit" of predicted in-hole pressure (dotted) using filtered surface sensor #2, File #2 data and 1-D (vertically averaged) model, superimposed on measured, filtered in-hole pressure data (solid) - U19ac.

Figure 5.67. "Best-fit" of predicted in-hole pressure (dotted) using filtered surface sensor #1. File #3 data and 1-D (vertically averaged) model, superimposed on measured, filtered in-hole pressure data (solid) - U19ac.

Figure 5.68. "Best-fit" of predicted in-hole pressure (dotted) using filtered surface sensor #2, File #3 data and 1-D (vertically averaged) model, superimposed on measured, filtered in-hole pressure data (solid) - U19ac.

Figure 5.69. "Best-fit" of predicted in-hole pressure (dotted) using filtered surface sensor #1, File #4 data and 1-D (vertically averaged) model, superimposed on measured, filtered in-hole pressure data (solid) - U19ac.

Figure 5.70. "Best-fit" of predicted in-hole pressure (dotted) using filtered surface sensor #2, File #4 data and 1-D (vertically averaged) model, superimposed on measured, filtered in-hole pressure data (solid) - U19ac.

Figure 5.71. "Best-fit" of predicted in-hole pressure (dotted) using filtered surface sensor #1, File #5 data and 1-D (vertically averaged) model, superimposed on measured, filtered in-hole pressure data (solid) - U19ac.

Figure 5.72. "Best-fit" of predicted in-hole pressure (dotted) using filtered surface sensor #2, File #5 data and 1-D (vertically averaged) model, superimposed on measured, filtered in-hole pressure data (solid) - U19ac.

diffusivities for U2fe. Table 5.11 shows the recomputed hydraulic diffusivities for U19ac.

TABLE 5.10

RECOMPUTED DIFFUSIVITIES AT NTS HOLE U2fe

Surface Sensor	Diffusivity (m ² /hr)	Uncertainty (m ² /hr)	Reduced Chi-Square
#1	5483	234	63
#2	5379	200	49

TABLE 5.11

RECOMPUTED DIFFUSIVITIES AT NTS HOLE U19ac

File No.	Surface Sensor	Diffusivity (m ² /hr)	Uncertainty (m ² /hr)	Reduced Chi-Square
#1	#1	101000	3796	40
	#2	84380	2155	22
#2	#1	56990	2870	16
	#2	63600	2244	70
#3	#1	64280	739	1.8
	#2	67350	596	1.0
#4	#1	47450	1188	34
	#2	50790	1121	25
#5	#1	24100	490	16
	#2	24200	480	15

It is fairly clear that, although the uncertainty in hydraulic diffusivity has been reduced somewhat, the values of diffusivity have not been changed significantly by eliminating the tidal component in the analysis. Note also that the reduced chi-square has not changed significantly. This is due primarily to the fact that the zero-phase filtering operation reduces the effective data length equal to the filter length. Thus, what we gain in noise reduction we loose in data loss.

Figure 5.73 shows a multi-layer inversion solution evaluated for U19ac. Layer delineation was determined by the principle contacts of ash-fall and bedded tuffs, and with the rhyolite lava contact near the bottom of the hole. Porosities were constrained to values presented in the Site Characteristics Summary (Howard, 1982). Models were run assuming that (a) the bentonite treatment imposed an effective no-flow boundary at the locations indicated in Fig. 5.73 and (b) that the bentonite treatment had no effect on the hole face boundary. The case with no bentonite yielded values of 520 Darcies for the Timber Mountain Tuff, 205 Darcies for the Paintbrush Tuff, and 381 Darcies for the Rhyolite Lava. The reduced chi-square was slightly less at 81.

No multi-layer models were run on the filtered data.

5.4 Analysis of Data Taken at NT3 hole U20ai

Near the completion of this report, we were given additional barometric pressure data taken at U20ai, also located on the Pahute Mesa. For completeness, we present the preliminary analysis of this data here.

Emplacement hole U20ai was spudded on June 20, 1981 and completed to a depth of 656 m. on August 26, 1981. The hole is cased to a depth of 17.7 m. Drill bit diameter was 2.44 m diameter. Figures 5.74, 5.75, and 5.76 show the recorded surface data from sensors #1 and #1, and the recorded in-hole pressure data, respectively.

Data was analyzed using the 1-D (vertically averaged) flow model. Table 5.12 shows the results of this analysis.

TABLE 5.12
COMPUTED DIFFUSIVITY AT U20ai

Surface Sensor	Diffusivity (m ² /hr)	Uncertainty (m ² /hr)	Reduced Chi-Square (m ² /hr)
#1	14675	273	131
#2	15280	260	105

The reduced chi-square indicated above may be slightly elevated over the true minimum since the data was not edited for outliers. Figures 5.77 and 5.78 show the "best fit" using the 1-D (vertically averaged) model for surface sensor #1 and #2, respectively. Table 5.13 shows the computed permeabilities for U20ai assuming porosities of 0.1 and 0.01.

Figure S.73. Data inversion results on a three layer model using the finite-difference 2-D flow model - U193c.

note: Assumed porosities in this multilayer interpretation are 0.15 for the Timber mountain tuff, 0.17 for the Paintbrush tuff, and 0.13 for the Rhyolite lava (from Howard, 1984). These values are probably not representative of the open (connected) porosity in these fractured volcanics. Please see Chapter VII for a discussion.

Figure 5.74. Surface sensor #1 pressure record - U20ai

Figure 5.75. Surface sensor #2 pressure record - U20ai

Figure 5.76. In-hole sensor pressure record - U20a1

Figure 5.77. "Best-fit" of predicted in-hole pressure (dotted) using surface sensor #1 and 1-D (vertically averaged) model, superimposed on measured in-hole pressure data (solid) - U20ai.

Figure 5.78. "Best-fit" of predicted in-hole pressure (dotted) using surface sensor #2 and 1-D (vertically averaged) model, superimposed on measured in-hole pressure data (solid) - U20ai.

TABLE 5.13
COMPUTED PERMEABILITIES FOR U20a1

Surface Sensor	Porosity	Permeability (Darcies)	Uncertainty (Darcies)
#1	0.1	96	2
	0.01	9.6	0.2
#2	0.1	100	2
	0.01	10	0.2

CHAPTER VI

REEVALUATION OF PREVIOUS AIR INJECTION TESTS

In light of the rather large permeabilities inferred by the analyses described in Chapter V, we reevaluated the air injection tests previously carried out at the Nevada Test Site. As discussed in Chapter I, an air injection test at NTS hole U10am-5 in September, 1969, indicated a permeability of 9.5 Darcies for the alluvium (Pasternak and McMullen, 1970). A second set of air-injection tests were carried out at the same hole in July of 1971. These tests indicated an alluvium permeability between 8 and 16 Darcies (Morris, et al., 1971). Interpretation of data from the 1971 U10am-5 air-injection tests were analyzed by both analytic methods (Morris, et al., 1971) and numerical methods (Rossa and Morris, 1972).

Air injection tests at hole W9-ITS-aXY30.5, carried out in July of 1971, indicated permeabilities of 8.2 Darcies for the alluvium, 12.9 Darcies for Vitric tuffs, 4.0 Darcies for the Grouse Canyon ashfall tuff, and 2.5 Darcies for undifferentiated tuffs and tuffaceous sandstone (Snoeberger, et al., 1971).

6.1 Air Injection tests at U10am-5 - July, 1971

Tests on hole U10am-5 consisted of surface air compressors injecting air into the hole at several flow rates with the in-hole pressure response subsequently measured. Figure 6.1 shows the geometry at this hole. Table 6.1 shows the recorded data (Morris, et al., 1971):

TABLE 6.1

AIR INJECTION TESTS AT U10am-5

Date	Flow Rate (Million cfd)	Injection Time (min)	Downhole Pressure (psia)	Downhole Temperature (legs. F)
7/20/71	2.28	20	20.8	92
7/21/71	1.18	28	17.8	89
	2.16	26	21.7	92
	4.13	22	29.3	97
7/22/71	7.44	30	41.0	70
	8.76	18	46.4	82
	9.84	19	48.5	83
	10.08	24	50.6	82

Figure 6.1. Hole geometry at Ulqam-5.

Morris, et. al., analyzed the data based on the assumption that the flow had reached steady state. No attempt to justify this assumption was made in the report. The same assumption was made in the previous air injection tests documented by Pasternak and McMullen (1970). The equation relating gas injection flow rate to down-hole pressure increase for the steady-state analysis used in their interpretation is given by:

$$Q = \frac{19.8 h K T_b (P_{hs2} - P_{es2})}{\mu P_b T_b \ln(R_e/R_h)}$$

where:

Q = Rate of flow at P_b and T_b (ft³/day)
 K = Permeability (Darcies)
 h = Length of uncased hole (ft)
 μ = Gas Viscosity at T_b (0.0165 cP)
 P_b = Base pressure (14.7 psia)
 T_b = Temperature of air at bottom of hole (degs. R)
 T_b = Base temperature of air (degs. R)
 P_h = Pressure of air at bottom of hole (psia)
 P_e = Pressure of air at r = R_e (12.7 psia)
 R_h = Radius of uncased hole (0.51 ft.)
 R_e = "Radius of effect" (assumed to be 425 ft.)
 Ln = Natural logarithm

One solves for the permeability K in terms of all the other parameters given in the above equation. Note that this solution assumes that at some distance from the hole, the perturbation due to air injection is zero. Snoberger, et al. did not discuss reasoning behind their choice of 425 feet for the "radius of effect". This happened to be the value used in the 1969 tests (Pasternak and McMullen, 1970). The latter chose the midpoint depth of the uncased part of the hole, 425, to be the radius of effect "for lack of better information".

Derivation of this expression, and a discussion of its application, can be found elsewhere (ERCB, 1975). Using this expression, Morris, et al. calculated a permeability range of between 8 and 16 Darcies for this hole.

Documentation in this report did not confirm that the gas pressure field surrounding the hole had reached steady state within 20 minutes of start of injection. We therefore reinterpreted the above-tabulated data in terms of a transient pressure solution assuming, with no better information, that injection was carried out for 20 minutes for each of the tests. The appropriate expression for the transient test is given elsewhere (ERCB, 1975):

$$Q = \frac{2 h K (P_{hss2} - P_{oss2})}{1.417E06 \mu P_b Th \ln(A/(R_{hss2}))}$$

where:

$$A = \frac{(2.637E-04) 4 K t}{(1.781) \phi \mu c}$$

t = time after start of injection (hrs)
 C = gas compressibility (1/Pb)
 mu = gas viscosity (cp)
 K = permeability (md)
 Pb = base pressure (14.7 psia)
 Ph = measured in-hole pressure (psia)
 P0 = initial in-hole pressure (psia)
 Rh = hole radius (ft)

Note that no assumptions need be made regarding radius of effect in the above expression. Indeed, sqrt(A) is the radius of effect, which clearly depends on the flow parameters and time. Since permeability K appears in two places in the above expression, the solution was carried out iteratively using a microcomputer-based automatic equation solver. The results are tabulated as follows:

TABLE 6.2

REEVALUATED PERMEABILITIES AT NTS HOLE U10am-5

Date	Flow Rate (MMcf/d)	Permeability (Darcies)
7/20/71	2.28	11.49
7/21/71	1.18	10.51
	2.16	9.61
	4.13	7.97
7/22/71	7.44	6.29
	8.76	5.44
	9.84	5.61
	10.08	5.33

These values are consistent with those obtained by Morris, et al. (1971) and McKullen and Pasternak (1970). Of some interest is the marked permeability decrease with increased flow rate. This

decrease is probably due to turbulence near the wellbore, which subsequently increases drag. This has the effect of decreasing the "effective" permeability. Rossa, et al. (1972) carried out some numerical simulations to better describe the effect. In any event, the permeability at U10am-5, for low flow rates, is on the order of 10 or 12 Darcies.

6.2 Air Injection Tests at NTS Hole U9-ITS-eXY30.5

Figure 6.2 shows the generalized geologic cross section through hole U9-ITS-XY31. Documentation of the data on hole U9-ITS-eXY30.5 was sufficient for reanalysis. On this hole, several injection tests were carried out at different stages of hole cement filling. The first test was a totally open hole. The second, third, and fourth tests were on the same hole that had been cemented (plugged) from the bottom of the hole to various geologic contacts delineated by well logs. Surface compressors injected air at various flow rates and down-hole pressure increases were measured. Details of the experiment are discussed in some detail by Snoeberger, et al., 1971. Table 6.3 shows the measured data.

TABLE 6.3

AIR INJECTION TESTS AT NTS HOLE U9-ITS-eXY30.5

Date	Hole Depth (ft.)	Flow Rate (million scfd)	Downhole Pressure (psia)	Down-Hole Temperature (deg. F)
7/7/71	1195	5.64	18.25	93
	1195	6.77	18.75	98.5
7/8/71	1034	9.50	18.80	93
	1034	7.68	17.80	87
7/12/71	457	7.87	20.00	70
	457	9.50	20.50	71
	457	10.01	21.60	78
7/13/71	305	9.17	22.00	75
	305	11.76	24.00	75
	305	7.66	21.00	75
7/14/71	139	6.00	35.50	113
	139	6.72	38.00	113
	139	3.84	27.00	115

The air injection was over "relatively short times - approximately 20 min., sometimes less - for pressure buildup". It is most unfortunate that this information was not documented in a somewhat more complete fashion.

Figure 6.2. Geologic cross section through hole U9-ITS-XY31.

Using the transient pressure solution given above, permeability values were calculated and assigned to the open hole section at each stage of the cement filling of the wellbore. These values are given in Table 6.4

TABLE 6.4

AVERAGED PERMEABILITIES AT HOLE U9-ITS-aXY30.5

Date	Hole Depth (feet)	Permeability (Darcies)
7/7/71	1195	1.68
	1195	1.86
7/8/71	1034	3.12
	1034	3.07
7/12/71	457	4.68
	457	5.28
	457	4.74
7/13/71	305	6.28
	305	6.27
	305	6.03
7/14/71	139	2.61
	139	2.49
	139	3.32

Considering that the flow rate was varied for different tests on the same hole depth, the above results are remarkably consistent for a given hole depth and for the fact that we are only estimating the air injection time.

In order to unravel the above averaged permeability values to interval permeabilities, we assume (as did Snoberger, et al., 1971) that the average permeability is equal to the weighted sum of interval permeabilities:

$$K(\text{avg}) = (1/L) \sum_{i=1}^N (K_i H_i)$$

where the H_i are the interval thicknesses, the K_i are the interval permeabilities, and N is the number of intervals over which the average is computed. By carrying out this "deconvolution" algebraically, one can obtain the following:

TABLE 6.5

"DECONVOLVED" FORMATION PERMEABILITIES
 MTS HOLE U9-ITS-XY30.5

Interval No.	Geologic Description	Interval Thickness (ft)	Permeability (Darcies)
1	alluvium	139	2.81
2	vitric tuff	166	9.03
3	Grouse Canyon Ashfall	152	2.30
4	undifferentiated tuffs, tuffaceous sandstone	577	0.93

These values are consistent, but somewhat lower (as expected) than those obtained by Snoberger, et al. (1971) who used the steady-state interpretation. Their analysis yielded values of 8.2 Darcies for the alluvium, 12.9 Darcies for the vitric tuffs, 4.0 Darcies for the Grouse Canyon ashfall tuff, and 2.5 Darcies for undifferentiated tuffs and tuffaceous sandstone.

CHAPTER VII

DISCUSSION AND CONCLUSIONS

We have attempted in this report to establish the usefulness of the barometric pressure testing method for evaluating formation permeability at the Nevada Test Site. Flow models have been developed from the most simple analytic and semi-analytic to quite sophisticated numerical techniques. These models have been used to evaluate barometric pressure transient taken at NTS holes U2fe, completed in the alluvium in Area 9, and U19ac and U20ai, completed in the Area 19 and Area 20 volcanics at Pahute Mesa.

Due to the fact that data interpretation on layered half-space models was found to be extremely computationally intensive, only a limited number of computer runs were made for multiple layer models. As a result, considerable use has been made of the 1-D (vertically averaged) flow model, which has been shown to be an excellent approximation to the 2-D (radial and vertical) flow model for a homogeneous and isotropic half-space. Values of hydraulic diffusivity based on this model should only be considered a first-order estimate of the diffusivity. However, comparing multiple layer models to the 1-D (vertically averaged) model for the same data set indicates that the simpler model for the homogeneous and isotropic half-space reflects the bulk properties of the entire section from the surface to the bottom of the hole.

It is also quite clear, from the analysis presented, that the estimated permeability for the formation intersected by the hole is significantly affected by near-surface permeability reduction due to pore blockage or "skin" caused by rain or snow, respectively. In order to better account for this effect, the semi-analytic 1-D (vertically averaged) model should have a near-surface skin term introduced analogous to conventional reservoir pressure transient "skin" term due to wellbore damage. The alternative would simply be to add a thin near-surface layer in the finite-difference model. The fact that these effects are clearly present in the data does not alter the basic conclusions of this report. Values of hydraulic diffusivity and permeability presented herein can be considered a lower bound on the true diffusivity and permeability.

We have also established in this work that there can be significant pressure signal contamination at the tidal frequencies. This contamination is primarily in the form of wellhead heating effects. The surface casing and seal represent an excellent conductor of heat energy from the surface to the hole. The slight increase in gas temperature within the hole causes a slight increase in pressure. The transfer function for this pressure variation represents a totally different mechanism than the transfer function representing pressure diffusion through the formation, and unless accounted for in an

interpretation scheme, represents a noise contamination of the in-hole pressure signal. This heat energy is in the form of absorbed radiant energy at the principle solar tide frequencies. Because of this contamination, it is advised that the pressure signals (both surface and in-hole) be low-pass filtered to remove this contamination prior to interpretation. One would, however, wish to maintain this higher frequency information for the purpose of gaining a greater resolving power on multiple layer model interpretation. One straight-forward procedure to eliminate, or at least reduce, this thermal contamination would be to either insulate the surface casing and seal (either by encapsulating them in sprayed foam insulation or simply burying under 3 to 4 feet of soil) or to encase the latter in a temperature-controlled environment.

Computed permeability using the barometric pressure analysis for U2fe, completed in the alluvium in Area 2, was found range from 30 Darcies to as high as 80 Darcies, depending on the model used for interpretation. These results are somewhat higher than permeability values previously reported for the alluvium. The latter range from 2.8 Darcies to 8.2 Darcies using air injection testing in Areas 9 and 10 and between 0.8 Darcies and 60 Darcies using the barometric method in Area 9. The ranges reported reflect different models used for interpreting the data. Differences of this magnitude can most likely be attributed to: (1) site-to-site variation in alluvium permeability, (2) the estimate for connected porosity is too high in the barometric tests and (3): differences in effective averaging volume between conventional air injection tests and the barometric method. The barometric technique is sampling a larger formation volume because of the very low frequency range of the signal. Larger sampling volumes typically reflect higher permeabilities due principally to block-fracture scaling. A fracture zone some distance from the hole will not be seen by a short-term air injection test but likely be seen by either a long-term air injection test or by using the barometric method.

Gas permeabilities associated with volcanics in the Mesa areas presented in this and other reports are higher than values published for the same volcanic sequence underlying the alluvium in the playa regions at the Nevada Test Site. The latter have been reported to be 12.9 Darcies for vitric tuffs, 8.2 Darcies for the Grouse Canyon ashfall tuff, and 2.5 Darcies for undifferentiated tuffs and tuffaceous sandstone for Area 9 (Snoeberger, et al., 1971). These particular values are slightly high due to the assumption that the air injection test had reached steady state flow (see Chap. VI for a reinterpretation in terms of a transient model). Neither the steady state nor transient models for the air injection tests require an estimate of formation porosity.

Depending on the assumed value of open (connected) porosity, the Mesa permeabilities as determined by the barometric method may be as much as two orders of magnitude greater than permeabilities reported for volcanics underlying the alluvium in

playa regions of the NTS. We emphasize here, however, that these higher values of permeability in the Mesa volcanics as inferred by the barometric method assume values of porosity ranging from 0.1 to 0.2, based on reported values of porosity given in the Site Characterization Summaries of the holes investigated. These values of porosity are inferred from estimates of water content (neutron logs) and grain and bulk densities (gamma-gamma logs), and represent total porosity, both open (connected) and closed.

Principle flow conduits in the saturated zone below the water table on the Pahute Mesa are fractures (Blankennagel and Weir, 1973). It is expected that the same is true of the unsaturated zone above the water table. As a result, we suspect that, at least for the volcanics, values of 0.1 to 0.2 for connected porosity for the volcanics at the Pahute Mesa are at least an order of magnitude high. If a more reasonable value for connected porosity of 0.01 is used for the volcanics, values of permeability as calculated using the barometric technique range from 10 to 60 Darcies for U19ac and roughly 10 Darcies for U20a1, depending upon the model used for data interpretation. These values are consistent with values of permeability determined by groundwater pump tests on the Pahute mesa (Blankennagel and Weir, 1973), which range between 3 and 50 Darcies. The interpretation of the pump test data, using Jacob's method, does not require an estimate of connected porosity.

As stated previously, the values for permeability on the Pahute Mesa are somewhat higher than permeabilities reported for volcanics underlying the alluvium in playa regions of the NTS. We point out that the latter is represented in published form only at one site (Area 9: hole U9-ITS-eXY30.5). Of the three sites under consideration, namely U19ac and U20a1 on the Pahute Mesa and U9-ITS-eXY30.5 in Area 9, the difference is largest (by a factor of 5 to 10) between U19ac on the eastern flank of the Silent Canyon Caldera and U9-ITS-eXY30.5 in Area 9. This difference is likely due to differences in post-emplacement tectonic history of the volcanics for Mesa and Playa areas. The difference is less significant between U20a1 (centrally located in the Silent Canyon Caldera) and U9-ITS-eXY30.5.

Conventional pump test data taken on Pahute Mesa indicate that the greatest permeabilities in the saturated zone below the water table occur in holes drilled along the east margin of the Silent Canyon Caldera (Blankennagel and Weir, 1973). The principle rock type in the saturated zone is rhyolite in this area. Values of permeability inferred from the barometric data on U19ac and U20a1 also suggest a greater permeability in the unsaturated zone on the east margin of the Silent Canyon Caldera (15 to 60 Darcies for U19ac and 10 Darcies for U20a1: both using an assumed connected porosity of 0.01 and using the 1-D vertically averaged flow model). The Silent Canyon Caldera is masked by younger volcanic rocks except along the east boundary, where the wall of the caldera is exposed in some localities. This ring fracture zone has experienced displacements between 5000 feet and 7000 feet (Blankennagel and Weir, 1973). Enhanced permeability is

expected on the eastern margin of the caldera due to both (1) proximity to the ring fracture structure and associated tectonic activity and (2) the extensive rhyolitic lava flows extruded along east margin faults. Rhyolites exhibit the largest permeabilities in the saturated zones of this area (Blankennagel and Weir, 1973).

The one multi-layer model (see Fig. 5.73) evaluated for U19ac indicated permeabilities of 668 Darcies for the Timber Mountain Tuff, 201 Darcies for the Paintbrush Tuff, and 330 Darcies for the Rhyolite lava (assuming porosities of 0.15, 0.17, and 0.13, respectively, as reported in the Site Characterization Summary). Normalizing these values to a much more reasonable value of 0.01 for connected porosity, one obtains permeabilities of 44.5 Darcies for the Timber Mountain Tuff, 13.4 Darcies for the Paintbrush Tuff, and 22 Darcies for the Rhyolite lava. These values are consistent in magnitude but somewhat inconsistent in formation specificity with the results of Blankennagel and Weir (1973), who found that the rhyolites on the Pahute Mesa yielded the largest permeability values.

We point out that during the drilling of U19ac, a significant drilling fluid loss was experienced over two zones within the Timber Mountain Tuff, one over a 150 m interval, suggesting significant pervasive fracturing in this section. Also, at the request of the author, Mr. Jeff Waggoner (geologist, LLNL) pulled the borehole TV log and checked for visual indication of fractures in the 50-100 m section below the casing in the Timber Mountain Tuff. He found a significant amount of fracturing in this area. Therefore, the results of the multi-layer inversion may be consistent with the actual vertical permeability distribution at this site. Further multi-layer models should be run to resolve the uniqueness of this solution.

It has been previously documented (Blankennagel and Weir, 1973) that, at least below the water table, permeability on the Pahute Mesa is very site specific. The vertical and horizontal distribution of intracaldera ash-fall and ash-flow tuffs and lava flows over short distances is variable. These abrupt changes in rock types make subsurface correlations between test holes extremely difficult. Because of this spatial variability, it is recommended that some form of formation permeability testing, either barometric or air injection, be carried out on every emplacement hole in the Mesa areas of the Nevada Test Site. Statistical methods such as variogram analysis and kriging could be used to predict inter-hole parameters and uncertainties to optimize future emplacement hole location selection within the area.

One critical question that has not been completely resolved in this report, primarily due to the large amount of computer time required to carry out a multiple layer inversion, is that of the uniqueness of the multiple layer interpretations. In the event that the computational effort cannot be reduced by optimizing the code, other means of constraining the solution must be found. One

method that would supply sufficient constraints on the solution is by measuring the relative mass flow rates into and out of the open hole as a function of depth. This can be achieved by instrumenting the hole (in addition to the high precision quartz pressure gauges) with a string of 5 to 10 high precision thermistors on a wireline the full length of the hole. Connected in a bridge-type circuit, currently available thermistors can be extremely sensitive to the cooling effect of any kind of gas flow. With this additional instrumentation, there would be left little room for question as to which formations or fractured areas were accepting the most flow. In an inversion-type scheme, both radial gas flow at the wellbore and in-hole pressure data could be fitted by the multi-layer model presented in this report.

In addition to the above, another alternative which is strongly recommended is to drill several slim holes in the Mesa areas for the purpose of barometric pressure testing. These holes, no greater than say a half a kilometer from each other, should each be instrumented with several high precision quartz pressure gauges, which are grouted in but still have hydraulic communication with the adjacent formation. If possible, it would be very useful to also have air injection ports in the column located between the pressure gauges for single-hole vertical permeability or tracer tests or hole-to-hole horizontal permeability or tracer tests. The combined use of the slim holes for air-injection tests would allow for independent checks on the data interpretation. The nature of the completion of this instrumentation, which allows only for 1-D vertical flow (ignoring lateral changes in formation permeability), constrains data interpretation much more than the open-hole analysis presented in this report. Barometric pressure data interpretation would be carried out throughout the year in order to establish variations introduced by the near-surface saturated zone.

There has historically been an emphasis on establishing permeability and not diffusivity at the Nevada Test Site. We again state that it is the diffusivity and not permeability that governs radioactive gas transport from the shot point to the surface. Thus, the question of connected porosity and how it relates to permeability is somewhat peripheral to the real problem of radioactive leaks. The large values of hydraulic diffusivity estimated at U19ac and U20ai on the Payute Mesa clearly are reason for concern regarding migration of radioactive gas from the shot point to the surface. We recommend that an analysis be carried to evaluate the effects of atmospheric pumping as a driving force for gas transport to the surface, as contrasted with simple molecular diffusion. We suspect that atmospheric pumping may play a major role in this transport. This analysis would tie in very well with the slim-hole tracer tests mentioned above.

One may also wish to consider modifying permeability about the shot point by means of polymer injection prior to the shot. Polymers can be tailored to have time-dependent viscosity. As a result, during pumping, the polymer can be tailored to have a low

viscosity to enhance migration through fractures surrounding the shot point, and then "set up" to form a barrier to gas transport. This is clearly an interesting, albeit possibly impractical, approach to the problem at hand. However, it warrants at least a study at the feasibility level considering the potential alternatives.

REFERENCES

Baker, J., Morris, G., and Snoeberger, D., "Drill Hole Permeability Tests - Instruments and Data Transmission", Univ. of Calif., Lawrence Livermore National Laboratory, UCID-15946, November 17, 1971

Bevington, P.R., "Data Reduction and Error Analysis for the Physical Sciences", McGraw-Hill, New York, 1969

Blankennagel, R.K. and Weir, J.E., Jr., "Geohydrology of the Eastern Part of Pahute Mesa, Nevada Test Site, Nye County, Nevada", U.S.G.S. Prof. Paper 712-B, 1973

Crump, K.S., "Numerical Inversion of Laplace Transforms Using a Fourier Series Approximation", Jour. ACM, 23, 1, January 1976, pp. 89-96

Dake, L.P., "Fundamentals of Reservoir Engineering", Elsevier, New York, 1982.

Godin, G., "The Analysis of Tides", Univ. of Toronto Press, 1972

Hanson, J.M., "In-Situ Fracture Characterization - A New Approach Using Multi-Frequency Flow Testing", Proceedings 24th U.S. Symposium on Rock Mechanics, Texas A&M Univ., June 20-23, 1983

Hanson, J.M., "Nonlinear Inversion of Pressure Transient Data", SPE/DOE/GRI 12848, Proceedings Unconventional Gas Recovery Symposium, Pittsburgh, May, 1984

Hanson, J.M., "Evaluation of Subsurface Fracture Geometry Using Fluid Pressure Response to Solid Earth Tidal Strain", Univ. of Calif., Lawrence Livermore National Laboratory, UCID-20156, September, 1984

Howard, N.W., "U2fe - Site Characteristics Summary", Univ. of Calif., Lawrence Livermore National Laboratory, Internal Report, November 1, 1982

Howard, N.W., "U19ac - Preliminary Site Characteristics Summary", Univ. of Calif., Lawrence Livermore National Laboratory, Internal Report, July 18, 1984

Morris, G.A., D.P. Snoeberger, C.J. Morris, and V.L. DuVal, "Field Measurements of Permeability of NTS Alluvium in Area 10", Univ. of Calif., Lawrence Livermore National Laboratory, UCID-15919, October 11, 1971

Morris, G. and Snoeberger, D., "Calculations of Pressure Change at Depth in a Nuclear Chimney Following Atmospheric Pressure Change", Univ. of Calif., Lawrence Livermore National Laboratory, UCID-15963, December 17, 1971

ECRB, "Theory and Practice of the Testing of Gas Wells", Energy Resources Conservation Board, Calgary, Canada, 1975, 3rd ed.

Fosternek, A. and E.T. McMullen, "Field Measurement of the Permeability of NTS Alluvium", Univ. of Calif., Lawrence Livermore National Laboratory, UCID-15727, July 28, 1970

Prickett, T.A. and Lonquist, C.G., "Selected Digital Computer Techniques for Groundwater Resource Evaluation", Illinois State Water Survey, Urbana, Bull. 55, 1971

Rozsa, R.B. and G.A. Morris, "Numerical Simulation of Air-Injection Experiments in NTS Alluvium", Univ. of Calif., Lawrence Livermore National Laboratory, UCID-15981, January 15, 1972

Rozsa, R.B., Snoeberger, D.F., and Baker, J., "Chimney Permeability Data Analysis", Univ. of Calif., Lawrence Livermore National Laboratory, UCID-16440, January 30, 1974

Rozsa, R.B., Snoeberger, D.F., and Baker, J., "Analysis of Atmospheric Pressure Response Data in Hole U9ITSu-29#2, Nevada Test Site", Univ. of Calif., Lawrence Livermore National Laboratory, UCID-16566, July 15, 1974

Rozsa, R., Snoeberger, D., and Baker, J., "Permeability of a Nuclear Chimney and Surface Alluvium, Area 2, ERDA NTS", Univ. of Calif., Lawrence Livermore National Laboratory, UCID-16722, March 10, 1975

Sherwood, A.E., "In-Situ Permeability Analysis using Barometric Pressure Fluctuations", Univ. of Calif., Lawrence Livermore National Laboratory, UCRL-51687, November 4, 1974

Snoeberger, D.F., "Chimney Permeability by Atmospheric Pressure Change - Test Start-Up", Univ. of Calif., Lawrence Livermore National Laboratory, UCID-16110, October 16, 1972

Snoeberger, D.F., Baker, J., and Morris, C.J., "Measurements and Correlation Analysis for Nuclear Chimney Permeability", Univ. of Calif., Lawrence Livermore National Laboratory, UCID-16302

Snoeberger, D.F., Baker, J., Morris, C.J., and Rozsa, R.B., "Permeability of a Nuclear Chimney and Surface Alluvium at the AEC Nevada Test Site", Univ. of Calif., Lawrence Livermore National Laboratory, UCID-16479, April 17, 1974

Snoeberger, D.F., Morris, C.J., and Baker, J., "Chimney Permeability by Atmospheric Pressure Change - Instruments and Data Handling", Univ. of Calif., Lawrence Livermore National Laboratory, UCID-16154, November 21, 1972

Snoeberger, D.F., Morris, C.J., and Baker, J., "Data and Results, Nuclear Chimney Permeability - Test 3", Univ. of Calif., Lawrence Livermore National Laboratory, UCID-16436, January 24, 1974

Snoeberger, D.F., Morris, C.J., and Morris, G.A., "Field Measurement of Permeabilities in NTS Area 9", Univ. of Calif., Lawrence Livermore National Laboratory, UCID-15895, August 23, 1971

Snow, D.T., "Rock Fracture Spacing Openings and Porocities", Am. Soc. Civil Engineers, Proc. Jour. Soil Mech. and Found. Div., 94, 1, 1968 pp. 73-91

Stehfest, H., "Numerical Inversion of Laplace Transforms", Comm. ACM 13, 47, Jan. 1970

Weeks, E.P., "Field Determination of Vertical Permeability to Air in the Unsaturated Zone", U.S.G.S. Prof. Paper 1051, 1978

Weeks, E.P., "Barometric Fluctuations in Wells Tapping Deep Unconfined Aquifers", Water Res. Research, 15, 5, 1979, pp. 1167-1176

COMPUTER PROGRAM: CAVITY

CAVITY computes the pressure response of a finite-length hole intersecting a homogeneous and isotropic half-space due to either a Heaviside step function or delta function applied pressure at the Earth's surface. The hole is partially cased and sealed at the surface. Method of calculation is semi-analytical.

C)

```

1 *LARGE
2 *NOFLOATCALLS
3 $DEBUG
4     PROGRAM CAVITY
5 C
6 C     WRITTEN BY:
7 C     JONATHAN M. HANSON
8 C     LAWRENCE LIVERMORE NATIONAL LABORATORY
9 C     1982
10 C     CDC CYBER 7600
11 C
12 C     MODIFICATIONS:
13 C     BY JONATHAN M. HANSON
14 C     TERRA TEK RESEARCH
15 C     SALT LAKE CITY, UTAH
16 C     IBM-PC
17 C
18 C     CAVITY SOLVES THE 2-D (RADIAL AND VERTICAL) FLOW DIFFUSION
19 C     EQUATION FOR A PARTIALLY CASED VERTICAL HOLE EMBEDDED IN A
20 C     HOMOGENEOUS AND ISOTROPIC HALF-SPACE. PRESSURE RESPONSE IN THE
21 C     HOLE IS CALCULATED GIVEN AN APPLIED SURFACE BAROMETRIC PRESSURE.
22 C     THIS SURFACE PRESSURE CAN BE A STEP FUNCTION OR DELTA FUNCTION
23 C     IN TIME. IN ADDITION TO THE CAVITY (HOLE) PRESSURE, THE RADIAL
24 C     PRESSURE GRADIENT IS CALCULATED ON THE HOLE WALL.
25 C
26 C     METHOD OF CALCULATION IS SEMI-ANALYTIC. A FREDHOLM INTEGRAL
27 C     EQUATION USING A NEUMANN GREEN'S FUNCTION AS PART OF THE
28 C     KERNEL IS SOLVED IN LAPLACE TRANSFORM SPACE FOR THE LAPLACE
29 C     TRANSFORM OF THE RADIAL PRESSURE GRADIENT AT THE WALL FACE. THE
30 C     PRESSURE GRADIENT, AND SUBSEQUENTLY THE PRESSURE, IS INVERTED TO
31 C     THE TIME DOMAIN USING A NUMERICAL LAPLACE INVERSION SCHEME.
32 C
33     DIMENSION A(200,50),X(100,10),D(100),T(100),B(100,100),Y(10)
34     DIMENSION DP(500),V(10),P(100),GAMMA(200)
35     DIMENSION ALPHA(200),P0(500),R(100),H(500,10),Z(100)
36     DIMENSION DPAW(100),LP(100),G(500,10),BETA(200),DEPTH(500)
37     DIMENSION XI(200),SQT(200),ILINE(2),JLINE(2),NNN(2)
38     DIMENSION XXX(2,2000),YYY(2,2000)
39 C
40     REAL MU
41 C
42     ID1=0
43     ID2=0
44     PI=4.0*ATAN(1.0)
45 C
46     NZ2=500
47     NZ3=30
48 C
49 C     MU IS VISCOSITY OF AIR (PG.SEC)
50 C     P0 IS ATMOSPHERIC PRESSURE (PA)
51 C
52     MU=2.0E-05
53     PATM=8.5E04
54 C
55     WRITE(*,490)
56     READ(*,*) D1,D2,XL,RW
57 C
58 C     IS THE SURFACE PRESSURE A STEP FUNCTION OR A DELTA FUNCTION?
59     WRITE(*,500)
60     READ(*,700) RSPTPE
61 C
62     DZ3=XL/FLOAT(NZ3-1)
63 C

```

```

64      10 WRITE(*,520)
65      READ(*,*) NF,NZ
66 C
67      20 DZ=(D2-D1)/FLOAT(NZ-1)
68      DZ2=XL/FLOAT(NZ2-1)
69 C
70 C      SET UP ALPHA ARRAY
71      DO 30 J=1,NF
1 72      ALPHA(J)=(FLOAT(J-1)+0.5)*PI/XL
1 73      30 CONTINUE
74 C
75 C      SET UP NODE ARRAY ON Z=ID1,D2). OPEN HOLE SECTION BELOW CASING
76      DO 40 J=1,NZ
1 77      Z(J)=FLOAT(J-1)*DZ+D1
1 78      40 CONTINUE
79 C
80 C      SET UP NODE ARRAY ON Z=I0,L1, SURFACE TO NO-FLOW BOUNDARY
81      KK1=0
82      KK2=0
83      DO 60 J=1,NZ2
1 84      DEPTH(J)=FLOAT(J-1)*DZ2
1 85      IF(KK1.NE.0) GOTO 50
1 86      IF(DEPTH(J).GT.D1) KK1=J
1 87      50 IF(KK2.NE.0) GOTO 60
1 88      IF(DEPTH(J).GE.D2) KK2=J
1 89      60 CONTINUE
90 C
91      DO 80 J=1,NF
1 92      DO 70 K=1,NZ
2 93      A(J,K)=SIN(ALPHA(J)*Z(K))
2 94      70 CONTINUE
1 95      80 CONTINUE
96 C
97      WRITE(*,530)
98      READ(*,*) NLAP
99 C
100 C      INITIALIZE NUMERICAL LAPLACE INVERSION
101 C      COEFFICIENT ARRAY V
102      CALL LINV(NLAP,V)
103 C
104 C      SPECIFY TIME HISTORY OR SPATIAL DISTRIBUTION CALCULATION
105      WRITE(*,590)
106      READ(*,700) JTYPE
107      IF(JTYPE.EQ.'T') GOTO 90
108      IF(JTYPE.EQ.'S') GOTO 110
109 C
110      90 WRITE(*,540)
111      READ(*,*) TIME1,TIME2
112      WRITE(*,550)
113      READ(*,*) NT
114 C
115      100 DT=(TIME2-TIME1)/FLOAT(NT-1)
116      GOTO 120
117 C
118      110 WRITE(*,600)
119      READ(*,*) TIME1
120      NT=1
121      DT=0.0
122 C
123 C      SPECIFY FORMATION POROSITY AND PERMEABILITY
124      120 WRITE(*,560)
125      READ(*,*) PHI,XK
126 C
127 C      CALCULATE FORMATION DIFFUSIVITY
128      DIFF=(XK*PATM)/(MU*PHI)
129 C

```

```

127 50000 FORMAT(/, ' ENTER OUTPUT FILENAME')
128 60000 FORMAT(A)
129 70000 FORMAT(/, ' ANOTHER TIME? (Y/N)')
130 71000 FORMAT(/, ' INPUT NLAP ((=0)')
131      END

132      SUBROUTINE HOMOG1(NLAYER, TIME, AMP)
133 C
134      COMMON /ROCK/XK(10), DIFF(10), Z(10), NUM(10), ZZ(1000)
135      COMMON /LAPL/NLAP, V(20)
136 C
137      DIMENSION A(18,18), AA(10,10), BB(10,10)
138      DIMENSION X(18), AMP(1000), AMP2(1000,10)
139      DIMENSION C(20,20), LP(40)
140 C
141 C
142 C      HOMOG1 COMPUTES THE PRESSURE AMPLITUDE RESPONSE OF A
143 C      LAYERED HALF-SPACE DUE TO A PRESCRIBED PRESSURE VARIATION
144 C      AT THE SURFACE OF A FORM:
145 C
146 C           $P(t) = L'[function]$ 
147 C
148 C      WHERE  $L'$  IS THE NUMERICAL LAPLACE INVERSE OPERATOR AND
149 C      function IS THE SURFACE PRESSURE IN LAPLACE SPACE.
150 C
151 C      PARAMETERS ARE:
152 C
153 C      NLAYER    NUMBER OF LAYERS
154 C      Z(J)      DISTANCE (M) FROM SURFACE TO BOTTOM OF JTH LAYER
155 C               IE.  $Z(J+1)-Z(J) = H(J+1)$ , WHERE H IS LAYER
156 C               THICKNESS. J IS LAYER INDEX (J=1 IS TOP LAYER)
157 C      XK(J)     PERMEABILITY ( $m^2$ ) OF THE JTH LAYER
158 C      DIFF(J)   DIFFUSIVITY ( $m^2/hr$ ) OF THE JTH LAYER
159 C
160 C
161 C      DIMENSION OF SYSTEM OF EQUATIONS
162      N=2*NLAYER-2
163 C
164      AB=ALOG(2.0)/TIME
165 C
166      DO 1650 KK=1,NLAP
167      S=FLOAT(KK)*AB
168 C
169 C      PRESCRIBE LAPLACE TRANSFORM OF SURFACE PRESSURE
170 C      IN THIS PARTICULAR CASE, THE SURFACE PRESSURE IS THE
171 C      HEAVISIDE STEP FUNCTION. 0 FOR  $T < 0$  AND 1 FOR  $T = 0$ .
172      FUN=1.0/S
173 C
174 C      ZERO OUT ARRAYS
175      DO 200 J=1,N
176      X(J)=0.0
177      AA(J,KK)=0.0
178      BB(J,KK)=0.0
179      DO 100 K=1,N
180      A(J,K)=0.0
181      100      CONTINUE
182      200      CONTINUE
183 C
184 C      EVALUATE ELEMENTS OF "A" ARRAY
185      DO 900 J=1,N
186      JM2=J-2
187      JM1=J-1
188      JP1=J+1
189      JP2=J+2
190 C

```



```

130 C      CHANGE DIFFUSIVITY FROM DARCY/SECONDS TO M**2/HOUR
131      DIFF=3.553686E-09*DIFF
132 C
133      FAC=2.0*DIFF*PHI/RW
134      DO 460 J=1,NT
135      IF(JTYPE.EQ.'T') GOTO 130
136 C
137      TT=TIME1
138      GOTO 140
139 C
140      T(J)=TIME1+FLOAT(J-1)*DT
141      TT=T(J)
142 C
143      140 AA=ALOG(2.0)/TT
144 C
145 C
146      DO 320 K=1,NLAP
147      ARG=FLOAT(K)*AA
148 C
149 C      SET UP GAMMA ARRAY (WEIGHTING FUNCTION IN GREEN'S FUNCTION
150 C      MATRIX
151      DO 150 LL=1,NF
152      SGT(LL)=SQRT(ALPHA(LL)*ALPHA(LL)+ARG/DIFF)
153      ZAP=SGT(LL)*RW
154      XK0=BESK0(ZAP)
155      XK1=BESK1(ZAP)
156      GAMMA(LL)=XK0/(ZAP*XK1)
157      150 CONTINUE
158 C
159 C      SET UP HOMOGENEOUS (IE. "NO-HOLE") SOLUTION P0
160      DO 160 L=1,N
161      ZZ=Z(L)
162      P0(L)=HOMOG(ARG,DIFF,ZZ,XL,RSPTPE)
163      R(L)=P0(L)
164      160 CONTINUE
165 C
166 C      SET UP B MATRIX, WHERE B=F/P0, F=DP/DR AT R=RW
167 C      CONFIGURE MATRIX ACCORDING TO TRAPEZOIDAL RULE
168      DO 190 M=1,NZ
169 C
170 C      FILL IN ROWS
171      DO 180 N=1,NZ
172      XMUL1=1.0
173      IF((N.EQ.1).OR.(N.EQ.NZ)) XMUL1=0.1
174 C      FILL IN COLUMNS
175      SUM=0.0
176      DO 170 N3=1,NF
177      SUM=SUM+GAMMA(N3)*A(N3,M)*A(N3,N)
178      170 CONTINUE
179      B(M,N)=(2.0*RW/XL)*SUM+FAC/(ARG*DZ)
180      B(M,N)=DZ*XMUL1*B(M,N)
181      180 CONTINUE
182      190 CONTINUE
183 C
184      NZP1=NZ+1
185 C      FILL IN RIGHT HAND SIDE (COLUMN NZ+1) WITH HOMOGENEOUS SOLN.
186      DO 195 MM=1,NZ
187      B(MM,NZP1)=R(MM)
188      195 CONTINUE
189 C
190 C      INVERT MATRIX B USING HIGH PRECISION CROUT ALGORITHM
191      EPS=0.000001
192      CALL CROUT(100,NZ,1,B,EPS,DET,IER,LP)
193
194 C
195 C      COEFFICIENTS OF DP/DR (X) AND DP/DR (Y) ARE STORED IN

```

```

2 195 C SAVE COEFFICIENTS OF DP/DR IN LAPLACE TRANSFORM SPACE IN
2 196 C X ARRAY, COEFFICIENTS OF CAVITY PRESSURE (IN LAPLACE TRANSFORM
2 197 C SPACE) IN Y ARRAY
2 198 C
2 199 DO 200 JJ=1,NZ
3 200 R(JJ)=B(JJ,NZP1)
3 201 X(JJ,K)=R(JJ)
3 202 200 CONTINUE
2 203 C
2 204 C COMPUTE FOURIER COEFFICIENTS FOR CAVITY EFFECT ON PRESSURE
2 205 C COLUMN ON (D1,D2)
2 206 DO 220 ML=1,NF
3 207 SUM=0.0
3 208 DO 210 MK=1,NZ
4 209 SUM=SUM+R(MK)*A(ML,MK)
4 210 210 CONTINUE
3 211 XI(ML)=DZ*SUM
3 212 BETA(ML)=GAMMA(ML)*XI(ML)
3 213 220 CONTINUE
2 214 C
2 215 IF(JTYPE.NE.'S') GOTO 250
2 216 C
2 217 C COMPUTE PRESSURE EFFECT OF CAVITY ON COLUMN (0,1) AND ADD
2 218 C TO HOMOGENEOUS SOLUTION
2 219 DO 240 ML=1,NZ2
3 220 SUM=0.0
3 221 DO 230 MK=1,NF
4 222 SUM=SUM+BETA(MK)*SIN(ALPHA(MK)*DEPTH(ML))
4 223 230 CONTINUE
3 224 H(ML,K)=HOMOG(ARG,DIFF,DEPTH(ML),XL,RSPTPE)
3 225 G(ML,K)=H(ML,K)-2.0*RW*SUM/XL
3 226 240 CONTINUE
2 227 GOTO 320
2 228 C
2 229 C COMPUTE PRESSURE EFFECT OF CAVITY ON COLUMN (D1,D2) AND ADD
2 230 C TO HOMOGENEOUS SOLUTION
2 231 250 DO 270 ML=1,NZ
3 232 SUM=0.0
3 233 DO 260 MK=1,NF
4 234 SUM=SUM+BETA(MK)*SIN(ALPHA(MK)*Z(ML))
4 235 260 CONTINUE
3 236 H(ML,K)=HOMOG(ARG,DIFF,Z(ML),XL,RSPTPE)
3 237 G(ML,K)=H(ML,K)-2.0*RW*SUM/XL
3 238 270 CONTINUE
2 239 C
2 240 320 CONTINUE
1 241 C
1 242 IF(JTYPE.EQ.'T') GOTO 370
1 243 C
1 244 C INVERT DP/DR AND P ON (D1,D2) AT THE CAVITY WALL (R=RW) INTO
1 245 C THE TIME DOMAIN (FROM LAPLACE SPACE)
1 246 DO 340 NN=1,NZ
2 247 FA1=0.0
2 248 DO 330 MM=1,NLAP
3 249 FA1=FA1+X(NN,MM)*V(MM)
3 250 330 CONTINUE
2 251 DPDFW(NN)=AA*FA1
2 252 XXX(1,NN)=Z(NN)
2 253 YYY(1,NN)=DPDFW(NN)
2 254 340 CONTINUE
1 255 C
1 256 ILINE(1)=1
1 257 JLINE(1)=7
1 258 NNN(1)=NZ
1 259 CALL XPLOT(NNN,1,ILINE,JLINE,XXX,YYY)
1 260 C
1 261 C INVERT P AND DP/DR ON (0,1) AT CAVITY WALL (R=RW) INTO TIME

```

```

1 262 C      DOMAIN (FROM LAPLACE SPACE)
1 263      DO 360 NN=1,NZ2
2 264      FA3=0.0
2 265      FA4=0.0
2 266      DO 350 MM=1,NLAP
3 267      FA3=FA3+G(NN,MM)*V(MM)
3 268      FA4=FA4+H(NN,MM)*V(MM)
3 269      350 CONTINUE
2 270      DP(NN)=AA*FA3
2 271      P0(NN)=AA*FA4
2 272      360 CONTINUE
1 273 C
1 274      NNN(1)=NZ2
1 275      NNN(2)=NZ2
1 276      ILINE(1)=1
1 277      ILINE(2)=1
1 278      JLINE(1)=7
1 279      JLINE(2)=7
1 280 C
1 281      DO 365 NN=1,NZ2
2 282      XXX(1,NN)=DEPTH(NN)
2 283      XXX(2,NN)=XXX(1,NN)
2 284      YYY(1,NN)=DP(NN)
2 285      YYY(2,NN)=P0(NN)
2 286      365 CONTINUE
1 287      CALL XPLOT(NNN,2,ILINE,JLINE,XXX,YYY)
1 288      GOTO 470
1 289 C
1 290 C      INVERT P AND P0 ON (D1,D2) AT THE CAVITY WALL (R=RW) INTO THE
1 291 C      TIME DOMAIN (FROM LAPLACE SPACE)
1 292      370 DO 390 NN=1,NZ
2 293      FA3=0.0
2 294      FA4=0.0
2 295      DO 380 MM=1,NLAP
3 296      FA3=FA3+G(NN,MM)*V(MM)
3 297      FA4=FA4+H(NN,MM)*V(MM)
3 298      380 CONTINUE
2 299      DP(NN)=AA*FA3
2 300      P0(NN)=AA*FA4
2 301      390 CONTINUE
1 302 C
1 303 C      AVERAGE HOMOGENEOUS SOLN. AND FULL SOLN. ON (D1,D2)
1 304      400 CALL SIMP(1.0,DP,ANS1,NZ)
1 305      CALL SIMP(1.0,P0,ANS2,NZ)
1 306 C
1 307      P(J)=ANS1/FLOAT(NZ)
1 308      Q(J)=ANS2/FLOAT(NZ)
1 309      XXX(1,J)=TT
1 310      XXX(2,J)=TT
1 311      YYY(1,J)=P(J)
1 312      YYY(2,J)=Q(J)
1 313
1 314      WRITE(*,620) TT
1 315      WRITE(*,630) P(J),Q(J)
1 316 C
1 317      460 CONTINUE
1 318 C
1 319      ILINE(1)=1
1 320      JLINE(1)=7
1 321      ILINE(2)=0
1 322      JLINE(2)=0
1 323      NNN(1)=NT
1 324      NNN(2)=NT
1 325      CALL XPLOT(NNN,2,ILINE,JLINE,XXX,YYY)
1 326 C
1 327      470 CONTINUE

```

```

327 740 CONTINUE
328 C
329 490 FORMAT(/, ' ENTER CASING LENGTH (M), HOLE DEPTH (M)',
330 X /, ' DEPTH TO NO-FLOW BOUNDARY (M), AND HOLE RADIUS (M)')
331 500 FORMAT(/, ' CALCULATE STEP RESPONSE (S) OR IMPULSE RESPONSE (I)')
332 520 FORMAT(/, ' ENTER NO. FOURIER COEFFICIENTS IN SERIES',
333 X /, ' AND NO. OF CAVITY NODES')
334 530 FORMAT(/, ' ENTER NUMBER OF TERMS RETAINED IN INVERSION',
335 X /, ' 4 IS A GOOD STARTING NUMBER')
336 540 FORMAT(/, ' ENTER START TIME (HRS) AND STOP TIME (HRS)')
337 550 FORMAT(/, ' ENTER NO. TIME POINTS')
338 560 FORMAT(/, ' ENTER POROSITY, PERMEABILITY (DARCYS)')
339 590 FORMAT(/, ' TIME (T) OR SPACE (S) CALCULATION?')
340 600 FORMAT(/, ' ENTER TIME (HRS)')
341 620 FORMAT(/, ' TIME (HRS) =', E12.4)
342 630 FORMAT(/, ' CAVITY PRESSURE FROM 2-D CALCULATION =', E12.4)
343 X /, ' AVERAGE OF 1-D CALCULATION OVER OPEN HOLE =', E12.4)
344 700 FORMAT(A)
345 C
346 740 STOP
347 END

```

```

348 FUNCTION BESK0(X)
349 A1=-0.57721566
350 A2=0.42278420
351 A3=0.23069756
352 A4=0.03488590
353 A5=0.00262698
354 A6=0.00010750
355 A7=0.00000740
356 C
357 B1=1.25331414
358 B2=-0.07832358
359 B3=0.02189568
360 B4=-0.01062446
361 B5=0.00587872
362 B6=-0.00251540
363 B7=0.00053208
364 C
365 IF (X.GT.2.0) GOTO 10
366 R=(X/2.0)**2
367 F=R*(A2+R*(A3+R*(A4+R*(A5+R*(A6+R*A7))))
368 BESK0=-ALOG(X/2.0)*BESJ0(X)+A1+F
369 RETURN
370 10 R=2.0/X
371 F=R*(B2+R*(B3+R*(B4+R*(B5+R*(B6+R*B7))))
372 BESK0=EXP(-X)*(B1+F)/SQRT(X)
373 RETURN
374 END

```

```

375 FUNCTION BESK1(X)
376 A1=0.15443144
377 A2=-0.67278579
378 A3=-0.18156897
379 A4=-0.01919402
380 A5=-0.00110404
381 A6=-0.00004686
382 C
383 B1=1.25331414
384 B2=0.23498619
385 B3=-0.03655620
386 B4=0.01504268
387 B5=-0.00780353
388 B6=0.00325614
389 B7=-0.00006245

```

```

390 C
391 IF (X.GT.2.0) GOTO 10
392 R=(X/2.0)**2
393 F=R*(A1+R*(A2+R*(A3+R*(A4+R*(A5+R*A6))))
394 BESK1=(1.0+F*X*ALOG(X/2.0)*BES11(X))/X
395 RETURN
396 10 R=2.0/X
397 F=R*(B2+R*(B3+R*(B4+R*(B5+R*(B6+R*B7))))
398 BESK1=(B1+F)*EXP(-X)/SQRT(X)
399 RETURN
400 END

401 FUNCTION BES10(X)
402 A1=3.5156229
403 A2=3.0899424
404 A3=1.2067492
405 A4=0.2659732
406 A5=0.0360768
407 A6=0.0045813
408 C
409 B1=0.39894228
410 B2=0.01328592
411 B3=0.00225319
412 B4=-0.00157565
413 B5=0.00916281
414 B6=-0.02057706
415 B7=0.02625537
416 B8=-0.01647633
417 B9=0.00392377
418 C
419 T=X/3.75
420 IF (X.GT.3.75) GOTO 10
421 R=T*T
422 F=R*(A1+R*(A2+R*(A3+R*(A4+R*(A5+R*A6))))
423 BES10=1.0+F
424 RETURN
425 10 R=1.0/T
426 F=R*(B2+R*(B3+R*(B4+R*(B5+R*(B6+R*(B7+R*(B8+R*B9))))
427 BES10=EXP(X)*(B1+F)/SQRT(X)
428 RETURN
429 END

430 FUNCTION BES11(X)
431 A1=0.87890594
432 A2=0.51498869
433 A3=0.15084934
434 A4=0.02656733
435 A5=0.00301532
436 A6=0.00032411
437 C
438 B1=0.39894228
439 B2=-0.03989424
440 B3=0.00362018
441 B4=0.00163801
442 B5=-0.01031555
443 B6=0.02282967
444 B7=-0.02895312
445 B8=0.01787654
446 B9=-0.00420059
447 C
448 T=X/3.75
449 IF (X.GT.3.75) GOTO 10
450 R=T*T
451 F=R*(A1+R*(A2+R*(A3+R*(A4+R*(A5+R*A6))))

```

```

451      (CONTINUE) RETURN
452      BES1=(0.5+F)*X
453      RETURN
454 10 R=1.0/T
455      F=R*(B2+R*(B3+R*(B4+R*(B5+R*(B6+R*(B7+R*(B8+R*B9))))))
456      BES1=EXP(X)*(B1+F)/SQRT(X)
457      RETURN
458      END

```

```

459      FUNCTION HOMOG(S,DIFF,Z,XL,NTYPE)
460 C      IF NTYPE=5, STEP RESPONSE
461 C      IF NTYPE=1, IMPULSE RESPONSE
462 C
463      IF(NTYPE.EQ.'S') FAC=1.0/S
464      IF(NTYPE.EQ.'I') FAC=1.0
465 C
466      ARG=SQRT(S/DIFF)
467      A1=ARG*(XL-Z)
468      A2=ARG*XL
469 C
470      HOMOG=FAC*(EXP(A1)+EXP(-A1))/(EXP(A2)+EXP(-A2))
471      RETURN
472      END

```

```

473      SUBROUTINE LINV(N,V)
474      DIMENSION V(50),G(100),H(100)
475 C
476 C      LINV COMPUTES THE COEFFICIENT ARRAY V USED FOR A NUMERICAL
477 C      LAPLACE TRANSFORM INVERSION
478 C
479 C      N = SOME EVEN INTEGER OF THE ORDER OF THE NUMBER OF DIGITS
480 C          THE MACHINE IS CAPABLE OF WORKING WITH
481 C
482      G(1)=1.0
483      NH=N/2
484      DO 10 I=1,N
1 485      G(I+1)=G(I)*FLOAT(I)
1 86 10 CONTINUE
487 C
488      H(1)=2.0/G(NH)
489      DO 20 I=2,NH
1 490      H(I)=(FLOAT(I)**NH)*G(2*I+1)/(G(NH-I+1)*G(I+1)+G(I))
1 491 20 CONTINUE
492 C
493      SN=2.0*FLOAT(MOD(NH,2))-1.0
494      DO 40 I=1,N
1 495      V(I)=0.0
1 496      NTOP=MIN0(I,NH)
1 497      NBOT=(I+1)/2
1 498      DO 30 K=NBOT,NTOP
2 499      V(I)=V(I)+H(K)/(G(I-K+1)+G(2*K-I+1))
2 500 30 CONTINUE
1 501 C
1 502      V(I)=SN*V(I)
1 503      SN=-SN
1 504 40 CONTINUE
505 C
506      RETURN
507      END

```

```

508      SUBROUTINE CROUT(MR,NR,NCC,M,ZCCL,D,ISRR,LP)
509 C      CROUT (1) OPERATES ON A COEFFICIENT MATRIX TO SOLVE A SYSTEM OF
510 C      SIMULTANEOUS EQUATIONS AND TO COMPUTE AN INVERSE AND (2)
511 C      COMPUTES DETERMINANT

```

```

511 C      CROUT REDUCES THE ORIGINAL MATRIX AND RIGHT-HAND SIDES UNTIL
512 C      UPON COMPLETION THE REDUCED MATRIX REPLACES THE ORIGINAL
513 C      MATRIX AND THE SOLUTIONS REPLACE THE RIGHT-HAND SIDES.
514 C
515 C
516      DIMENSION A(1), LP(1)
517      IF(NR.GT.MR) GO TO 210
518      MTX = MR*NR
519      MRA = MR + 1
520      MRS = MR - 1
521      MDN = MR - NR
522      MTR = MTX - MDN
523      MTRA = MTX + 1
524      DT = 1.
525      IERR = 0
526      NRS = NR - 1
527      DO 2 I = 1,NR
1 528          LP(I) = I
1 529      2  CONTINUE
530      IF(NCC)210,605,1001
531      805  NTC = NR + NR
532          MTT = MR*NTC - MDN
533          J = MTRA
534          DO 19 K = MTRA,MTT,MR
1 535              JF = K + NRS
1 536              DO 18 KX = K,JF
2 537                  A(KX) = 0.
2 538      18  CONTINUE
1 539          A(J) = 1.
1 540          J = J + MRA
1 541      19  CONTINUE
542          GO TO 1
543      1001  NTC = NR + NCC
544          MTT = MR*NTC - MDN
545      1  IF(NTC.LE.NR) GO TO 210
546          DO 70 I = 1,NR
1 547              IS = I - 1
1 548              II = MR+IS + 1
1 549              IISB = II - 1
1 550              IIAD = II + MR
1 551              ICF = MR+I - MDN
1 552              ICS = ICF - NRS
1 553              IIA = II + 1
1 554              TEMP = 0.
1 555              DO 31 J = II,MTT,MR
556                  IF(I.EQ.1) GO TO 33
557                  KF = J - 1
2 558                      KS = J - I + 1
2 559                      KX = I
2 560                      DO 30 K = KS,KF
3 561                          A(J) = A(J) - A(KX)*A(K)
3 562                          KX = KX + MR
3 563      30  CONTINUE
2 564      33  IF(J.GT.MTR) GO TO 31
2 565          IF(ABS(A(J)).LE.TEMP) GO TO 31
2 566          TEMP = ABS(A(J))
2 567          NX = J/MR + 1
2 568      31  CONTINUE
569          IF(I.EQ.NR) GO TO 35
1 570          IF(NX.EQ.1) GO TO 35
571          ITEMP = LP(NX)
572          LP(NX) = LP(I)
1 573          LP(I) = ITEMP
1 574          LPS = MR*NX - MRS
1 575          DO 34 K = ICS,ICF
576              TEMP = A(K)
577          34  A(K) = A(K)*LPS

```

```

1 577      DATA = 0.000000
2 578      A(LPIS) = TEMP
2 579      LPIS = LPIS + 1
2 580 34    CONTINUE
1 581      DT = - DT
1 582 35    DT = DT*A(II)
1 583      IF(ZMCH -ABS(A(II))) 45,45,220
1 584 45    DO 46 J = IIAD,MTT,MR
2 585      A(J) = A(J)/A(II)
2 586 46    CONTINUE
1 587      IF(1.EQ.1) GO TO 70
1 588      IF(1.EQ.NR) GO TO 78
1 589      DO 47 M = 1,ICF
2 590      KX = M - ICF + 1
2 591      DO 48 KY = 1,ICL,ISB
3 592      A(M) = A(M) - A(KX) + A(KY)
3 593      KX = KX + MR
3 594 48    CONTINUE
2 595 47    CONTINUE
1 596 70    CONTINUE
1 597 78    NRTAD = MTX + NR
1 598      DO 180 I = 1,NRS
1 599      IREV = NRTAD - I
1 600      KRS = IREV - MR+1
1 601      DO 170 IRCNT = IREV,MTT,MR
2 602      KCS = IRCNT + 1
2 603      DO 160 K = KRS,MTT,MR
3 604      A(IRCNT) = A(IRCNT) - A(-CS) + A(K)
3 605      KCS = KCS + 1
3 606 160    CONTINUE
2 607 170    CONTINUE
1 608 180    CONTINUE
1 609      DO 6 I = 1,NRS
1 610 9      IF(LP(I).EQ.1) GO TO 6
1 611      NX = LP(I)
1 612      LP(I) = LP(NX)
1 613      LP(NX) = NX
1 614      IXS = MTX + 1
1 615      IY = MTX + NX
1 616      DO 7 IX = IXS,MTT,MR
2 617      TEMP = A(IX)
2 618      A(IX) = A(IY)
2 619      A(IY) = TEMP
2 620      IY = IY + MR
2 621 7      CONTINUE
1 622      GO TO 9
1 623 6      CONTINUE
1 624      RETURN
1 625 210    IERR = 2
1 626      NTC = NR
1 627      DT = 9E + 10
1 628      RETURN
1 629 220    IERR = 1
1 630      RETURN
1 631      END

```

```

632
633      SUBROUTINE SIMP(H,Y,ANS,NDIM)
634 C
635 C H = INTEGRATION STEP SIZE
636 C Y = INPUT VECTOR OF FUNCTION VALUES
637 C ANS = RETURNED VALUE OF INTEGRAL
638 C NDIM = DIMENSION OF Y
639 C Z = INTERNAL VECTOR OF INTEGRAL VALUES
640 C
641 C DIMENSION Y(50),Z(50)

```



```

642      HT=0.3333333333* $\pi$ 
643      IF (NDIM-5) 700,500,100
644 C
645 C NDIM IS GREATER THAN 5. PREPARATIONS OF INTEGRATION LOOP
646 100 SUM1=Y(2)+Y(2)
647      SUM1=SUM1+SUM1
648      SUM1=HT*(Y(1)+SUM1+Y(3))
649      AUX1=Y(4)+Y(4)
650      AUX1=AUX1+AUX1
651      AUX1=SUM1+HT*(Y(3)+AUX1+Y(5))
652      AUX2=HT*(Y(1)+3.875*(Y(2)+Y(5))+2.625*(Y(3)+Y(4))+Y(5)
653      SUM2=Y(5)+Y(5)
654      SUM2=SUM2+SUM2
655      SUM2=AUX2-HT*(Y(4)+SUM2+Y(6))
656      Z(1)=0.0
657      AUX=Y(3)+Y(3)
658      AUX=AUX+AUX
659      Z(2)=SUM2-HT*(Y(2)+AUX+Y(4))
660      Z(3)=SUM1
661      Z(4)=SUM2
662      IF (NDIM-6) 500,500,200
663 C
664 C INTEGRATION LOOP
665 200      DO 400 I=7,NDIM,2
1 666      SUM1=AUX1
1 667      SUM2=AUX2
1 668      AUX1=Y(I-1)+Y(I-1)
1 669      AUX1=AUX1+AUX1
1 670      AUX1=SUM1+HT*(Y(I-2)+AUX1+Y(I))
1 671      Z(I-2)=SUM1
1 672      IF (I-NDIM) 300,600,600
1 673 300      AUX2=Y(I)+Y(I)
1 674      AUX2=AUX2+AUX2
1 675      AUX2=SUM2+HT*(Y(I-1)+AUX2+Y(I+1))
1 676 400      Z(I-1)=SUM2
677 500 Z(NDIM-1)=AUX1
678      Z(NDIM)=AUX2
679      ANS=Z(NDIM)
680      RETURN
681 600 Z(NDIM-1)=SUM2
682      Z(NDIM)=AUX1
683      ANS=Z(NDIM)
684      RETURN
685 C END OF INTEGRATION LOOP
686 C
687 700 IF (NDIM-3) 1200,1100,800
688 C
689 C NDIM IS EQUAL TO 4 OR 5
690 800 SUM2=1.125*HT*(Y(1)+3.0*Y(2)+3.0*Y(3)+Y(4))
691      SUM1=Y(2)+Y(2)
692      SUM1=SUM1+SUM1
693      SUM1=HT*(Y(1)+SUM1+Y(3))
694      Z(1)=0.0
695      AUX1=Y(3)+Y(3)
696      AUX1=AUX1+AUX1
697      Z(2)=SUM2-HT*(Y(2)+AUX1+Y(4))
698      IF (NDIM-5) 1000,900,900
699 900 AUX1=Y(4)+Y(4)
700      AUX1=AUX1+AUX1
701      Z(5)=SUM1+HT*(Y(3)+AUX1+Y(5))
702 1000 Z(3)=SUM1
703      Z(4)=SUM2
704      ANS=Z(NDIM)
705      RETURN
706 C
707 C NDIM IS EQUAL TO 3

```

```

708 1100 SUM1=HT*(1.25*Y(1)+2.0*Y(2)-0.25*Y(3))
709      SUM2=Y(2)+Y(2)
710      SUM2=SUM2+SUM2
711      Z(3)=HT*(Y(1)+SUM2+Y(3))
712      Z(1)=0.0
713      Z(2)=SUM1
714      ANS=Z(NDIM)
715      RETURN
716 C
717 1200 IF (NDIM.EQ.1) GO TO 1300
718      Z(1)=0.0
719      Z(2)=(Y(1)+(Y(2)-Y(1))*0.5)+HT
720      ANS=Z(NDIM)
721      RETURN
722 C
723 1300 Z(1)=0.0
724      ANS=Z(NDIM)
725      RETURN
726      END

```

COMPUTER PROGRAM: HOMOG1
HOMOG1D

HOMOG1 and HOMOG1D compute the pressure response within a layered half-space given an applied surface Heaviside pressure step function increase. Each layer must be assigned a given porosity, thickness, and permeability. Method of calculation is semi-analytical.

C)

```

1 C      ++++++
2 C      +//
3 C      +//
4 C      +//
5 C      +//
6 C      +//
7 C      A COMPUTER PROGRAM WRITTEN FOR:
8 C      EARTH SCIENCE DEPARTMENT
9 C      UNIVERSITY OF CALIFORNIA
10 C     LAWRENCE LIVERMORE NATIONAL LABORATORY
11 C     LIVERMORE, CALIFORNIA
12 C
13 C     WRITTEN BY:
14 C     APPLIED GEOPHYSICS GROUP
15 C     TERRA TEK RESEARCH
16 C     420 WEST 3000 A WAY
17 C     SALT LAKE CITY, UTAH 84108
18 C
19 C     APPLICATION:
20 C     EVALUATION OF THE BAROMETRIC PRESSURE RESPONSE WITHIN A LAYERED
21 C     HALF-SPACE GIVEN A PRESCRIBED SURFACE BAROMETRIC PRESSURE. THE
22 C     SURFACE PRESSURE IS ASSUMED TO HAVE THE FORM:
23 C
24 C              $P(t) = L'[function]$ 
25 C
26 C     WHERE:  $L'$  IS THE NUMERICAL LAPLACE INVERSION OPERATOR AND function
27 C     IS THE SURFACE PRESSURE IN LAPLACE SPACE.
28 C
29 C
30 C     PRESSURE AMPLITUDES ARE CALCULATED AS FUNCTIONS OF DEPTH
31 C     FOR A GIVEN HALF-SPACE CONFIGURATION AND A GIVEN TIME. UP TO
32 C     10 LAYERS OF GIVEN THICKNESS, POROSITY, AND PERMEABILITY CAN BE
33 C     HANDLED (THIS IS EASILY MODIFIED BY CHANGING A FEW DIMENSION
34 C     STATEMENTS). 1-D (VERTICAL) FLOW IS ASSUMED, AND BOTH PRESSURE AND
35 C     FLUX CONTINUITY BOUNDARY CONDITIONS ARE SATISFIED AT EACH INTERFACE.
36 C     ALTHOUGH INPUT AS A FINITE THICKNESS LAYER, THE LOWER BOUNDARY OF
37 C     THE BOTTOM-MOST LAYER REALLY GOES TO INFINITY, WITH A ZERO-PRESSURE
38 C     BOUNDARY CONDITION THERE.
39 C
40 C     COMMENT: THE NUMBER OF NODES (AN INTERACTIVE REQUEST) DOES NOT EFFECT
41 C     SOLUTION PRECISION. THIS IS SIMPLY THE NUMBER OF LOCATIONS
42 C     THAT THE PRESSURE AMPLITUDE AND PHASE IS TO BE EVALUATED.
43 C
44 C     METHOD:
45 C     HIGH-PRECISION SEMI-ANALYTIC APPROACH USING THE CONCEPT OF
46 C     PROPAGATOR MATRICES.
47 C
48 C     $DEBUG
49 C     $STORAGE:2
50 C     $LARGE
51 C     $NOFLOATCALLS
52 C
53 C     PROGRAM DRIVER
54 C     COMMON /ROCK/XX(10),DIFF(10),Z(10),NJM(10),ZZ(1000)
55 C     COMMON /LAPL/NLAP,V(20)
56 C     DIMENSION AMP(1000),POR(10),H(10)
57 C
58 C     CHARACTER*20 FNAME
59 C
60 C     SET MS-FORTRAN 6207 MATH ERROR EXCEPTIONS

```

```

61 C      (MASK OVERFLOW AND UNDERFLOW)
62      CALL LCWRQ (4927)
63 C
64      WRITE (*,71000)
65      READ (*,*) NLAP
66 C
67 C      SET UP INVERSE LAPLACE TRANSFORM COEFFICIENT ARRAY
68      CALL LINV(NLAP,V)
69 C
70 C      SET VALUES FOR GAS VISCOSITY (c.p.sec) AND MEAN GAS PRESSURE (p.a)
71      VISC=2.0E-05
72      PAVG=8.5E04
73 C
74 C      INPUT HALF-SPACE PARAMETERS
75      WRITE (*,10000)
76      READ (*,*) NLAVER
77 C
78      KOUNT=0
79      SUM=0.0
80      DO 100 J=1,NLAVER
1 81          WRITE (*,20000) J
1 82          READ (*,*) H(J),XK(J),POR(J)
1 83 C
1 84          Z(J)=SUM+H(J)
1 85          SUM=Z(J)
1 86 C
1 87 C      CALCULATE LAYER DIFFUSIVITY
1 88          DIFF(J)=XK(J)*PAVG/(POR(J)*VISC)
1 89 C
1 90 C      CONVERT DIFFUSIVITY FROM M**2/SEC TO M**2/HR
1 91          DIFF(J)=3600.0*DIFF(J)
1 92 C
1 93          WRITE (*,40000) J
1 94          READ (*,*) NUM(J)
1 95          KOUNT=KOUNT+NUM(J)
1 96      100 CONTINUE
97 C
98      ZBOT=7*(NLAVER)
99 C
100 C
101      200 WRITE (*,50000)
102      READ (*,60000) FNAME
103      OPEN (2,FILE=FNAME, STATUS='NEW')
104 C
105      WRITE (*,30000)
106      READ (*,*) TIME
107 C
108      WRITE (2,*) KOUNT
109 C
110      CALL HOMOG1(NLAVER,TIME,AMP)
111 C
112      DO 300 J=1,KOUNT
1 113          WRITE (2,*) ZZ(J),AMP(J)
1 114      300 CONTINUE
115 C
116      CLOSE (2)
117      WRITE (*,70000)
118      READ (*,60000) FLAG
119      IF (FLAG.EQ.'Y') GO TO 300
120 C
121      STOP
122      10000 FORMAT(/,' INPUT NO. LAYERS ')
123      20000 FORMAT(/,' INPUT THICKNESS (m), PERMEABILITY (m**2), AND ',)
124      X ' POROSITY FOR LAYER ', I2)
125      30000 FORMAT(/,' INPUT TIME (hrs) ')
126      40000 FORMAT(/,' INPUT NO. NODES IN LAYER',I2)

```

```

127 50000 FORMAT(/, ' ENTER OUTPUT FILENAME' )
128 60000 FORMAT(A)
129 70000 FORMAT(/, ' ANOTHER TIME? (Y/N)' )
130 71000 FORMAT(/, ' INPUT NLAP (<=8)' )
131      END

132      SUBROUTINE HOMOG1(NLAYER,TIME,AMP)
133 C
134      COMMON /ROCK/XK(10),DIFF(10),Z(10),NUM(10),ZZ(1000)
135      COMMON /LAPL/NLAP,V(120)
136 C
137      DIMENSION A(10,10),AA(10,10),BB(10,10)
138      DIMENSION X(10),AMP(1000),AMP2(1000,10)
139      DIMENSION C(20,20),LP(40)
140 C
141 C
142 C      HOMOG1 COMPUTES THE PRESSURE AMPLITUDE RESPONSE OF A
143 C      LAYERED HALF-SPACE DUE TO A PRESCRIBED PRESSURE VARIATION
144 C      AT THE SURFACE OF A FORM:
145 C
146 C          P(t) = L' [function]
147 C
148 C      WHERE L' IS THE NUMERICAL LAPLACE INVERSE OPERATOR AND
149 C      function IS THE SURFACE PRESSURE IN LAPLACE SPACE.
150 C
151 C      PARAMETERS ARE:
152 C
153 C      NLAYER      NUMBER OF LAYERS
154 C      Z(J)        DISTANCE (M) FROM SURFACE TO BOTTOM OF JTH LAYER
155 C                  IE. Z(J+1)-Z(J) = H(J+1), WHERE H IS LAYER
156 C                  THICKNESS. J IS LAYER INDEX (J=1 IS TOP LAYER)
157 C      XK(J)       PERMEABILITY (m**2) OF THE JTH LAYER
158 C      DIFF(J)     DIFFUSIVITY (m**2/hr) OF THE JTH LAYER
159 C
160 C
161 C      DIMENSION OF SYSTEM OF EQUATIONS
162 C      N=2*NLAYER-2
163 C
164 C      AB=ALOG(2.0)/TIME
165 C
166 C      DO 1650 KK=1,NLAP
167 C          S=FLOAT(KK)*AB
168 C
169 C      PRESCRIBE LAPLACE TRANSFORM OF SURFACE PRESSURE
170 C      IN THIS PARTICULAR CASE, THE SURFACE PRESSURE IS THE
171 C      HEAVISIDE STEP FUNCTION. 0 FOR T < 0 AND 1 FOR T >= 0.
172 C      FUN=1.0/S
173 C
174 C      ZERO OUT ARRAY:
175 C          DO 200 J=1,N
176 C              X(J)=0.0
177 C              AA(J,KK)=0.0
178 C              BB(J,KK)=0.0
179 C              DO 100 K=1,N
180 C                  A(J,K)=0.0
181 C          CONTINUE
182 C      CONTINUE
183 C
184 C      EVALUATE ELEMENTS OF "A" ARRAY
185 C          DO 900 J=1,N
186 C              JM2=J-2
187 C              JM1=J-1
188 C              JP1=J+1
189 C              JP2=J+2
190 C

```

```

2 191          IF (MOD(J,2).EQ.0) GO TO 700
2 192 C
2 193 C      ODD ROWS OF A
2 194      500      L=(J+1)/2
2 195 C      NOTE: L IS LAYER NUMBER
2 196          LP1=L+1
2 197          ARG1=SQRT(S/DIFF(L))
2 198          ARG2=SQRT(S/DIFF(LP1))
2 199 C
2 200          A(J,J) = XK(L)*ARG1*EXP(ARG1*Z(L))
2 201          IF (L.EQ.1) A(J,J)=A(J,J)+XK(L)*ARG1*EXP(-ARG1*Z(L))
2 202          A(J,JP1)= XK(LP1)*ARG2*EXP(-ARG2*Z(L))
2 203          IF (L.EQ.NLAYER) GO TO 600
2 204          A(J,JP2)=-XK(LP1)*ARG2*EXP(ARG2*Z(L))
2 205          IF (L.EQ.1) GO TO 900
2 206      600      A(J,JM1)=-XK(L)*ARG1*EXP(-ARG1*Z(L))
2 207          GO TO 900
2 208 C
2 209 C      EVEN ROWS
2 210      700      L=J/2
2 211 C      NOTE: L IS LAYER NUMBER
2 212          LP1=L+1
2 213 C
2 214          ARG1=SQRT(S/DIFF(L))
2 215          ARG2=SQRT(S/DIFF(LP1))
2 216 C
2 217          A(J,JM1)= EXP(ARG1*Z(L))
2 218          IF (L.EQ.1) A(J,JM1)=A(J,JM1)-EXP(-ARG1*Z(L))
2 219          A(J,J)=- EXP(-ARG2*Z(L))
2 220          IF (L.EQ.NLAYER) GO TO 800
2 221          A(J,JP1)=-EXP(ARG2*Z(L))
2 222          IF (L.EQ.1) GO TO 900
2 223      800      A(J,JM2)= EXP(-ARG1*Z(L))
2 224      900      CONTINUE
1 225 C
1 226 C
1 227          DO 1300 J=1,N
2 228              DO 1200 K=1,N
3 229                  C(J,K)=A(J,K)
2 230      1200          CONTINUE
2 231      1300          CONTINUE
1 232 C
1 233 C      SET UP "RIGHT HAND SIDE" OF REAL MATRIX EQUATION
1 234          ARG=SQRT(S/DIFF(1))
1 235          F1=XK(1)*ARG*EXP(-ARG*Z(1))
1 236          F2=-EXP(-ARG*Z(1))
1 237          X(1)=F1
1 238          X(2)=F1+F2
1 239 C
1 240          NP1=N+1
1 241 C
1 242          DO 1400 J=1,N
2 243              C(J,NP1)=X(J)
2 244      1400          CONTINUE
1 245 C
1 246 C
1 247 C      SOLVE SET OF SIMULTANEOUS EQUATIONS
1 248 C      NOTE: "YENBOB" IS A COPYRIGHTED SUBROUTINE BY
1 249 C      PEERLESS ENGINEERING SERVICE FOR
1 250 C      JOHN WILEY AND SONS, INC.
1 251 C
1 252          CALL YENBOB(C,20,N)
1 253 C
1 254          DO 1500 J=1,N
2 255              X(J)=C(J,NP1)
2 256      1500          CONTINUE

```

```

1 257 C
1 258 C   CALCULATE LAYER COEFFICIENTS AA AND BB
1 259   BB(1, KK)=X(1)
1 260   AA(1, KK)=FUN-BB(1, KK)
1 261   DO 1600 J=2, N, 2
2 262     INDEX=(J/2)+1
2 263     AA(INDEX, KK)=X(J)
2 264 C
2 265 C   THE NEXT LINE INSURES THAT BB(NLAYER) IS ZERO.
2 266 C   REQUIRED FOR A FINITE SOLUTION AS Z GETS LARGE
2 267     IF (INDEX.EQ.NLAYER) GO TO 1600
2 268 C
2 269     BB(INDEX, KK)=X(J+1)
2 270 1600   CONTINUE
1 271 1650   CONTINUE
2 272 C
2 273 C   COMPUTE AMPLITUDE(Z).
2 274 C
2 275   DO 2100 KK=1, NLAP
1 276     S=FLOAT(KK)*AB
1 277 C
1 278     KOUNT=0
1 279     DO 2000 J=1, NLAYER
2 280       ARG=SDAT(S/DIFF(J))
2 281       IF (J.EQ.1) GO TO 1700
2 282 C
2 283 C   LAYER THICKNESS
2 284       H=Z(J)-Z(J-1)
2 285       ZTOP=Z(J-1)
2 286       GO TO 1800
2 287 C
2 288 1700     H=Z(J)
2 289     ZTOP=0.0
2 290 1800     DZ=H/LOAT(NUM(J)-1)
2 291 C
2 292 C   COMPUTE PRESSURE AT EACH DEPTH WITHIN LAYER
2 293 C   J
2 294     DO 1900 K=1, NUM(J)
3 295       KOUNT=KOUNT+1
3 296       ZZ(KOUNT)=FLOAT(K-1)*DZ+ZTOP
3 297       PP=AA(J, KK)*EXP(-ARG*ZZ(KOUNT))+BB(J, KK)*EXP(ARG*ZZ
3 298         (KOUNT))
3 299       AMP2(KOUNT, KK)=PP
3 300 1900     CONTINUE
2 301 2000     CONTINUE
1 302 2100     CONTINUE
303 C
304 C   COMPUTE INVERSE LAPLACE TRANSFORM AT EACH DEPTH
305   DO 2500 J=1, KOUNT
1 306     SUM=0.0
1 307     DO 2400 KK=1, NLAP
2 308       SUM=SUM+V(KK)*AMP2(J, KK)
2 309 2400     CONTINUE
1 310     AMP(J)=AB*SUM
1 311 2500     CONTINUE
312 C
313   RETURN
314   END

315   SUBROUTINE YENBOB(A, IRQW, ICOL)
316 C
317 C   PURPOSE...YENBOB FINDS THE SOLUTION OF A SYSTEM OF N EQUATIONS IN
318 C   N UNKNOWNNS.
319 C
320   DIMENSION A(1)

```



```

321      ICOL = ICOL + 1
322      DO 800 K = 1, ICOL
1 323      INDEX1 = (K-1)*IROW + K
1 324      A(INDEX1) = 1./A(INDEX1)
1 325      DO 200 J = 1, ICOL1
2 326      IF (J-K) 100, 200, 100
2 327      100 INDEX2 = (J-1)*IROW + K
2 328      A(INDEX2) = A(INDEX2)*A(INDEX1)
2 329      200 CONTINUE
1 330      DO 600 I = 1, ICOL
2 331      IF (I-K) 300, 600, 300
2 332      300 INDEX3 = (K-1)*IROW + I
2 333      DO 500 J = 1, ICOL1
3 334      IF (J-K) 400, 500, 400
3 335      INDEX2 = (J-1)*IROW + I
3 336      INDEX4 = (J - 1)*IROW + K
3 337      A(INDEX2) = A(INDEX2) - A(INDEX4)*A(INDEX3)
3 338      500 CONTINUE
2 339      600 CONTINUE
1 340      DO 800 I = 1, ICOL
2 341      IF (I-K) 700, 800, 700
2 342      700 INDEX2 = (K-1)*IROW + I
2 343      A(INDEX2) = -A(INDEX2)*A(INDEX1)
2 344      800 CONTINUE
345      RETURN
346      END

```

```

347      SUBROUTINE LINV(N,V)
348      DIMENSION V(50),G(100),H(100)
349 C
350      G(1)=1.0
351      NH=N/2
352      DO 100 I=1,N
1 353      G(I+1)=G(I)*FLOAT(I)
1 354      100 CONTINUE
355      H(1)=2.0/G(NH)
356      DO 200 I=2,NH
1 357      H(I)=(FLOAT(I)*NH)*G(2*I+1)/(G(NH-I+1)*G(I+1)*G(I))
1 358      200 CONTINUE
359      SN=2.0*FLOAT(MOD(NH,2))-1.0
360      DO 400 I=1,N
1 361      V(I)=0.0
1 362      NTOP=MIND(I,NH)
1 363      NBOT=(I+1)/2
1 364      DO 300 K=NBOT,NTOP
2 365      V(I)=V(I)+H(K)/(G(I-K+1)*G(2*K-I+1))
2 366      300 CONTINUE
1 367      V(I)=SN*V(I)
1 368      SN=-SN
1 369      400 CONTINUE
370      RETURN
371      END

```

C)

```

1 C      ****
2 C      *//
3 C      +//
4 C      +//
5 C      +//
6 C      *****
7 C      ***** DOUBLE PRECISION VERSION OF HOMOG1 *****
8 C
9 C      A COMPUTER PROGRAM WRITTEN FOR:
10 C     EARTH SCIENCE DEPARTMENT
11 C     UNIVERSITY OF CALIFORNIA
12 C     LAWRENCE LIVERMORE NATIONAL LABORATORY
13 C     LIVERMORE, CALIFORNIA
14 C
15 C     WRITTEN BY:
16 C     APPLIED GEOPHYSICS GROUP
17 C     TERRA TEK RESEARCH
18 C     420 WAKARA WAY
19 C     SALT LAKE CITY, UTAH 84108
20 C
21 C     APPLICATION:
22 C     EVALUATION OF THE BAROMETRIC PRESSURE RESPONSE WITHIN A LAYERED
23 C     HALF-SPACE GIVEN A PRESCRIBED SURFACE BAROMETRIC PRESSURE. THE
24 C     SURFACE PRESSURE IS ASSUMED TO HAVE THE FORM:
25 C
26 C              $P(t) = L'[function]$ 
27 C
28 C     WHERE  $L'$  IS THE NUMERICAL LAPLACE INVERSION OPERATOR AND function
29 C     IS THE SURFACE PRESSURE IN LAPLACE SPACE.
30 C
31 C
32 C     PRESSURE AMPLITUDES ARE CALCULATED AS FUNCTIONS OF DEPTH
33 C     FOR A GIVEN HALF-SPACE CONFIGURATION AND A GIVEN TIME. UP TO
34 C     10 LAYERS OF GIVEN THICKNESS, POROSITY, AND PERMEABILITY CAN BE
35 C     HANDLED (THIS IS EASILY MODIFIED BY CHANGING A FEW DIMENSION
36 C     STATEMENTS). 1-D (VERTICAL) FLOW IS ASSUMED, AND BOTH PRESSURE AND
37 C     FLUX CONTINUITY BOUNDARY CONDITIONS ARE SATISFIED AT EACH INTERFACE.
38 C     ALTHOUGH INPUT AS A FINITE THICKNESS LAYER, THE LOWER BOUNDARY OF
39 C     THE BOTTOM-MOST LAYER REALLY GOES TO INFINITY, WITH A ZERO-PRESSURE
40 C     BOUNDARY CONDITION THERE.
41 C
42 C     COMMENT: THE NUMBER OF NODES (AN INTERACTIVE REQUEST) DOES NOT EFFECT
43 C     SOLUTION PRECISION. THIS IS SIMPLY THE NUMBER OF LOCATIONS
44 C     THAT THE PRESSURE AMPLITUDE AND PHASE IS TO BE EVALUATED.
45 C
46 C     METHOD:
47 C     HIGH-PRECISION SEMI-ANALYTIC APPROACH USING THE CONCEPT OF
48 C     PROPAGATOR MATRICES.
49 C
50 C     $DEBUG
51 C     $LARGE
52 C     $NOFLOATCALLS
53 C
54 C     PROGRAM DRIVER
55 C     REAL*8 XK(10),DIFF(10),Z(10),ZZ(1000),V(20),
56 C     X AMP(1000),POR(10),H(10)
57 C     REAL*8 PAVG,SUM,TIME,VISC,ZBO
58 C     COMMON /ROCK/XK,DIFF,Z,NUM(10),ZZ
59 C     COMMON /LAPL/NLAP,V
60 C

```

```

61      CHARACTER*20 FNAME
62 C
63 C      SET MS-FORTRAN 8087 MATH ERROR EXCEPTIONS
64 C      (MASK OVERFLOW AND UNDERFLOW)
65      CALL LCWRQQ (4927)
66 C
67      WRITE (*,80000)
68      READ (*,*) NLAP
69 C
70 C      SET UP INVERSE LAPLACE TRANSFORM COEFFICIENT ARRAY
71      CALL LINV(NLAP,V)
72 C
73 C      SET VALUES FOR GAS VISCOSITY (pa.sec) AND MEAN GAS PRESSURE (pa)
74      VISC=2.0D-05
75      PAVG=8.5D04
76 C
77 C      INPUT HALF-SPACE PARAMETERS
78      WRITE (*,10000)
79      READ (*,*) NLayer
80 C
81      KOUNT=0
82      SUM=0.0D+00
83      DO 100 J=1,NLayer
1 84          WRITE (*,20000) J
1 85          READ (*,*) A1,A2,A3
1 86          H(J)=DBLE(A1)
1 87          XK(J)=DBLE(A2)
1 88          POR(J)=DBLE(A3)
1 89 C
1 90          Z(J)=SUM+H(J)
1 91          SUM=Z(J)
1 92 C
1 93 C      CALCULATE LAYER DIFFUSIVITY
1 94          DIFF(J)=XK(J)*PAVG/(POR(J)*VISC)
1 95 C
1 96 C      CONVERT DIFFUSIVITY FROM M**2/SEC TO M**2/HR
1 97          DIFF(J)=3600.0D+00*DIFF(J)
1 98 C
1 99          WRITE (*,40000) J
1 100         READ (*,*) NUM(J)
1 101         KOUNT=KOUNT+NUM(J)
1 102 100     CONTINUE
103 C
104         ZBOT=Z(NLayer)
105 C
106 C
107 200 WRITE (*,50000)
108     READ (*,60000) FNAME
109     OPEN (2,FILE=FNAME, STATUS='NEW')
110 C
111     WRITE (*,30000)
112     READ (*,*) TTT
113     TIME=DBLE(TTT)
114 C
115     WRITE (2,*) KOUNT
116 C
117     CALL HOMOG1(NLayer,TIME,AMP)
118 C
119     DO 300 J=1,KOUNT
1 120         A1=SNGL(ZZ(J))
1 121         A2=SNGL(AMP(J))
1 122         WRITE (2,*) A1,A2
1 123 300     CONTINUE
124 C
125     CLOSE (2)
126     WRITE (*,70000)

```

```

127      READ (*, 60000) FLAG
128      IF (FLAG.EQ.'Y') GO TO 200
129 C
130      STOP
131 10000 FORMAT(/,' INPUT NO. LAYERS ')
132 20000 FORMAT(/,' INPUT THICKNESS (m), PERMEABILITY (m**2), AND ',/,
133      X ' POROSITY FOR LAYER ', I2)
134 30000 FORMAT(/,' INPUT TIME (hrs) ')
135 40000 FORMAT(/,' INPUT NO. NODES IN LAYER',I2)
136 50000 FORMAT(/,' ENTER OUTPUT FILENAME')
137 60000 FORMAT(A)
138 70000 FORMAT(/,' ANOTHER TIME? (Y/N)')
139 80000 FORMAT(/,' INPUT NLAP (<=16)')
140      END

141      SUBROUTINE HOMOG1(NLAYER,TIME,AMP)
142 C
143      REAL*8 XK(10),DIFF(10),Z(10),ZZ(1000),V(20)
144      REAL*8 A(10,10),AA(10,20),BB(10,20)
145      REAL*8 X(10),AMP(1000),AMP2(1000,20)
146      REAL*8 C(20,20)
147      REAL*8 AB,ARG,ARG1,ARG2,DZ,F1,F2,FUN,H,PD,S,
148      X SUM,TIME,ZTOP
149 C
150      COMMON /ROCK/XK,DIFF,Z,NUM(10),ZZ
151      COMMON /LAPL/NLAP,V
152 C
153 C
154 C
155 C      HOMOG1 COMPUTES THE PRESSURE AMPLITUDE RESPONSE OF A
156 C      LAYERED HALF-SPACE DUE TO A PRESCRIBED PRESSURE VARIATION
157 C      AT THE SURFACE OF A FORM:
158 C
159 C       $P(t) = L'[function]$ 
160 C
161 C      WHERE L' IS THE NUMERICAL LAPLACE INVERSE OPERATOR AND
162 C      function IS THE SURFACE PRESSURE IN LAPLACE SPACE.
163 C
164 C      PARAMETERS ARE:
165 C
166 C      NLAYER    NUMBER OF LAYERS
167 C      Z(J)      DISTANCE (M) FROM SURFACE TO BOTTOM OF JTH LAYER
168 C               IE.  $Z(J+1)-Z(J) = H(J+1)$ , WHERE H IS LAYER
169 C               THICKNESS. J IS LAYER INDEX (J=1 IS TOP LAYER)
170 C      XK(J)     PERMEABILITY (m**2) OF THE JTH LAYER
171 C      DIFF(J)   DIFFUSIVITY (m**2/hr) OF THE JTH LAYER
172 C
173 C
174 C      DIMENSION OF SYSTEM OF EQUATIONS
175      N=2*NLAYER-2
176 C
177      AB=DLOG(2.0D+00)/TIME
178 C
179      DO 1300 KK=1,NLAP
180          S=DBLE(FLOAT(KK))*AB
181 C
182 C      PRESCRIBE LAPLACE TRANSFORM OF SURFACE PRESSURE
183 C      IN THIS PARTICULAR CASE, THE SURFACE PRESSURE IS THE
184 C      HEAVISIDE STEP FUNCTION, 0 FOR T ( 0 AND 1 FOR T ) = 0.
185      FUN=1.0D+00/S
186 C
187 C      ZERO OUT ARRAYS
188      DO 200 J=1,N
189          X(J)=0.0D+00
190          AC(J,KK)=0.0D+00

```

```

2 191          BB(J,KK)=0.0D+00
2 192          DO 100 K=1,N
3 193          A(J,K)=0.0D+00
2 194      100      CONTINUE
2 195      200      CONTINUE
1 196 C
1 197 C      EVALUATE ELEMENTS OF "A" ARRAY
1 198          DO 700 J=1,N
2 199          JM2=J-2
2 200          JM1=J-1
2 201          JP1=J+1
2 202          JP2=J+2
2 203 C
2 204          IF (MOD(J,2).EQ.0) GO TO 500
2 205 C
2 206 C      ODD ROWS OF A
2 207      300      L=(J+1)/2
2 208 C      NOTE: L IS LAYER NUMBER
2 209          LP1=L+1
2 210          ARG1=DSQRT(S/DIFF(L))
2 211          ARG2=DSQRT(S/DIFF(LP1))
2 212 C
2 213          A(J,J) = XK(L)*ARG1*DEXP(ARG1*Z(L))
2 214          IF (L.EQ.1) A(J,J)=A(J,J)+XK(L)*ARG1*DEXP(-ARG1*Z(L))
2 215          A(J,JP1)= XK(LP1)*ARG2*DEXP(-ARG2*Z(L))
2 216          IF (L.EQ.NLAYER) GO TO 400
2 217          A(J,JP2)=-XK(LP1)*ARG2*DEXP(ARG2*Z(L))
2 218          IF (L.EQ.1) GO TO 700
2 219      400      A(J,JM1)=-XK(L)*ARG1*DEXP(-ARG1*Z(L))
2 220          GO TO 700
2 221 C
2 222 C      EVEN ROWS
2 223      500      L=J/2
2 224 C      NOTE: L IS LAYER NUMBER
2 225          LP1=L+1
2 226 C
2 227          ARG1=DSQRT(S/DIFF(L))
2 228          ARG2=DSQRT(S/DIFF(LP1))
2 229 C
2 230          A(J,JM1)= DEXP(ARG1*Z(L))
2 231          IF (L.EQ.1) A(J,JM1)=A(J,JM1)-DEXP(-ARG1*Z(L))
2 232          A(J,J)=- DEXP(-ARG2*Z(L))
2 233          IF (L.EQ.NLAYER) GO TO 600
2 234          A(J,JP1)=-DEXP(ARG2*Z(L))
2 235          IF (L.EQ.1) GO TO 700
2 236      600      A(J,JM2)= DEXP(-ARG1*Z(L))
2 237      700      CONTINUE
1 238 C
1 239 C
1 240          DO 900 J=1,N
2 241          DO 800 K=1,N
3 242          C(J,K)=A(J,K)
2 243      800      CONTINUE
2 244      900      CONTINUE
1 245 C
1 246 C      SET UP "RIGHT HAND SIDE" OF REAL MATRIX EQUATION
1 247          ARG=DSQRT(S/DIFF(1))
1 248          F1=XK(1)*ARG*DEXP(-ARG*Z(1))
1 249          F2=-DEXP(-ARG*Z(1))
1 250          X(1)=FUN*F1
1 251          X(2)=FUN*F2
1 252 C
1 253          NP1=N+1
1 254 C
1 255          DO 1000 J=1,N
2 256          C(J,NP1)=X(J)

```

```

2 257 1000          CONTINUE
1 258 C
1 259 C
1 260 C          SOLVE SET OF SIMULTANEOUS EQUATIONS
1 261 C          NOTE: "YENBOB" IS A COPYRIGHTED SUBROUTINE BY
1 262 C          PEERLESS ENGINEERING SERVICE FOR
1 263 C          JOHN WILEY AND SONS, INC.
1 264 C
1 265          CALL YENBOB(C,20,N)
1 266 C
1 267          DO 1100 J=1,N
2 268          X(J)=C(J,NP1)
2 269 1100          CONTINUE
1 270 C
1 271 C          CALCULATE LAYER COEFFICIENTS AA AND BB
1 272          BB(1,KK)=X(1)
1 273          AA(1,KK)=FUN-BB(1,KK)
1 274          DO 1200 J=2,N,2
2 275          INDEX=(J/2)+1
2 276          AA(INDEX,KK)=X(J)
2 277 C
2 278 C          THE NEXT LINE INSURES THAT BB(NLAYER) IS ZERO.
2 279 C          REQUIRED FOR A FINITE SOLUTION AS Z GETS LARGE
2 280          IF (INDEX.EQ.NLAYER) GO TO 1200
2 281 C
2 282          BB(INDEX,KK)=X(J+1)
2 283 1200          CONTINUE
1 284 1300          CONTINUE
2 285 C
2 286 C          COMPUTE AMPLITUDE(Z).
2 287 C
2 288          DO 1800 KK=1,NLAP
1 289          S=DBLE(FLOAT(KK))*AB
1 290 C
1 291          KOUNT=0
1 292          DO 1700 J=1,NLAYER
2 293          ARG=DSQRT(S/DIFF(J))
2 294          IF (J.EQ.1) GO TO 1400
2 295 C
2 296 C          LAYER THICKNESS
2 297          H=Z(J)-Z(J-1)
2 298          ZTOP=Z(J-1)
2 299          GO TO 1500
2 300 C
2 301 1400          H=Z(J)
2 302          ZTOP=0.0D+00
2 303 1500          DZ=H/DBLE(FLOAT(NUM(J)-1))
2 304 C
2 305 C          COMPUTE PRESSURE AT EACH DEPTH WITHIN LAYER
2 306 C          J
2 307          DO 1600 K=1,NUM(J)
3 308          KOUNT=KOUNT+1
3 309          ZZ(KOUNT)=DBLE(FLOAT(K-1))*DZ+ZTOP
3 310          PC=AA(J,KK)*DEXP(-ARG*ZZ(KOUNT))+BB(J,KK)*DEXP(ARG*ZZ
3 311          X          (KOUNT))
3 312          AMP2(KOUNT,KK)=PP
3 313 1600          CONTINUE
2 314 1700          CONTINUE
1 315 1800          CONTINUE
2 316 C
2 317 C          COMPUTE INVERSE LAPLACE TRANSFORM AT EACH DEPTH
2 318          DO 2000 J=1,KOUNT
1 319          SUM=0.0D+00
1 320          DO 1900 KK=1,NLAP
2 321          SUM=SUM+V(KK)*AMP2(J,KK)
2 322 1900          CONTINUE

```

```

323      AMP(I)=ABSX6UM
324      2000      CONTINUE
325 C
326      RETURN
327      END

328      SUBROUTINE YENBOB(A,IRGW,ICOL)
329 C
330 C      PURPOSE...YENBOB FINDS THE SOLUTION OF A SYSTEM OF N EQUATIONS IN
331 C      N UNKNOWNNS.
332 C
333      REAL*8 A(1)
334      ICOL1 = ICOL + 1
335      DO 500 K = 1,ICOL
336      INDEX1 = (K-1)*IRGW + K
337      A(INDEX1) = 1.D+00/A(INDEX1)
338      DO 200 J = 1,ICOL1
339      IF (J-K) 100,300,100
340      100      INDEX2 = (J-1)*IRGW + K
341      A(INDEX2) = A(INDEX2)+A(INDEX1)
342      200      CONTINUE
343      DO 600 I = 1,ICOL
344      IF (I-K) 300,600,300
345      300      INDEX3 = (K-1)*IRGW + I
346      DO 500 J = 1,ICOL1
347      IF (J-K) 400,500,400
348      400      INDEX3 = (J-1)*IRGW + I
349      INDEX4 = (J - 1)*IRGW + K
350      A(INDEX2) = A(INDEX2) - A(INDEX4)*A(INDEX3)
351      500      CONTINUE
352      600      CONTINUE
353      DO 800 I = 1,ICOL
354      IF (I-K) 700,800,700
355      700      INDEX2 = (K-1)*IRGW + I
356      A(INDEX2) = -A(INDEX2)*A(INDEX1)
357      800      CONTINUE
358      RETURN
359      END

360      SUBROUTINE LINV(N,V)
361      REAL*8 V(20),G(100),H(100),SN
362 C
363      G(1)=1.D+00
364      NH=N/2
365      DO 100 I=1,N
366      G(I+1)=G(I)*DBLE(FLOAT(I))
367      100      CONTINUE
368      H(1)=2.D+00/G(NH)
369      DO 200 I=2,NH
370      H(I)=(DBLE(FLOAT(I))*NH)*G(2*I-1)/(G(NH-I+1)*G(I-1)+G(I))
371      200      CONTINUE
372      SN=2.D+00*DBLE(FLOAT(MOD(NH,2)))-1.D+00
373      DO 400 I=1,N
374      V(I)=0.D+00
375      NTOP=MIN0(I,NH)
376      NBOT=(I-1)/2
377      DO 300 K=NBOT,NTOP
378      V(I)=V(I)+H(K)/(G(I-K+1)*G(2*K-I+1))
379      300      CONTINUE
380      V(I)=SN*V(I)
381      SN=-SN
382      400      CONTINUE
383      RETURN

```

COMPUTER PROGRAM: HOMOG2

HOMOG2 computes the pressure response within a layered half-space given an applied surface sinusoidal pressure variation. Each layer must be assigned a given porosity, thickness, and permeability. Method of calculation is semi-analytical.

C)

```

1 C      ++++++
2 C      +//
3 C      +// HOMOGE
4 C      +//
5 C      ++++++
6 C
7 C      A COMPUTER PROGRAM WRITTEN FOR:
8 C      EARTH SCIENCE DEPARTMENT
9 C      UNIVERSITY OF CALIFORNIA
10 C     LAWRENCE LIVERMORE NATIONAL LABORATORY
11 C     LIVERMORE, CALIFORNIA
12 C
13 C     WRITTEN BY:
14 C     APPLIED GEOPHYSICS GROUP
15 C     TERRA TEK RESEARCH
16 C     420 WAKARA WAY
17 C     SALT LAKE CITY, UTAH 84108
18 C
19 C     APPLICATION:
20 C     EVALUATION OF THE BAROMETRIC PRESSURE RESPONSE WITHIN A LAYERED
21 C     HALF-SPACE GIVEN A TIME-VARYING SURFACE BAROMETRIC PRESSURE. THE
22 C     SURFACE PRESSURE IS ASSUMED TO HAVE THE FORM:
23 C
24 C            $P(t) = \exp(i\omega t)$ 
25 C
26 C     WHERE  $t$  IS TIME,  $\omega$  IS ANGULAR FREQUENCY, AND  $i$  IS  $\text{SQRT}(-1)$ .
27 C     THUS, THE SURFACE PRESSURE AMPLITUDE IS 1.0
28 C
29 C     PRESSURE AMPLITUDE AND PHASE ARE CALCULATED AS FUNCTIONS OF DEPTH
30 C     FOR A GIVEN HALF-SPACE CONFIGURATION AND A GIVEN FREQUENCY. UP TO
31 C     10 LAYERS OF GIVEN THICKNESS, POROSITY, AND PERMEABILITY CAN BE
32 C     HANDLED (THIS IS EASILY MODIFIED BY CHANGING A FEW DIMENSION
33 C     STATEMENTS). 1-D (VERTICAL) FLOW IS ASSUMED, AND BOTH PRESSURE AND
34 C     FLUX CONTINUITY BOUNDARY CONDITIONS ARE SATISFIED AT EACH INTERFAC
35 C     ALTHOUGH INPUT AS A FINITE THICKNESS LAYER. THE LOWER BOUNDARY O
36 C     THE BOTTOM-MOST LAYER REALLY GOES TO INFINITY, WITH A ZERO-PRESS
37 C     BOUNDARY CONDITION THERE.
38 C
39 C     COMMENT: THE NUMBER OF NODES (AN INTERACTIVE REQUEST) DOES NOT E
40 C     SOLUTION PRECISION. THIS IS SIMPLY THE NUMBER OF LOCATI
41 C     THAT THE PRESSURE AMPLITUDE AND PHASE IS TO BE EVALUATED.
42 C
43 C     METHOD:
44 C     HIGH-PRECISION SEMI-ANALYTIC APPROACH USING THE CONCEPT OF
45 C     PROPAGATOR MATRICES.
46 C
47 C     *CEBUG
48 C     *STORAGE:2
49 C     *NOFLDCALLS
50 C
51 C     PROGRAM DRIVER
52 C     COMMON /ROCK/XX(10),DIFF(10),Z(10),NUM(10),ZZ(1000)
53 C     DIMENSION AMP(1000),PHASE(1000),POR(10),H(10)
54 C
55 C     CHARACTER*20 FNAME
56 C
57 C     SET MS-FORTRAN 8087 MATH ERROR EXCEPTIONS
58 C     (MASK OVERFLOW AND UNDERFLOW)
59 C     CALL LCWR00 (4327)
60 C

```

```

61      PI=3.14159265
62 C
63 C      SET VALUES FOR GAS VISCOSITY (Pa.sec) AND MEAN GAS PRESSURE (Pa)
64      VISC=2.0E-05
65      PAVG=8.5E04
66 C
67 C      INPUT HALF-SPACE PARAMETERS
68      WRITE (*,10000)
69      READ (*,*) NLayer
70 C
71      KOUNT=0
72      SUM=0.0
73      DO 100 J=1,NLayer
74          WRITE (*,20000) J
75          READ (*,*) H(J),XK(J),POR(J)
76 C
77          Z(J)=SUM+H(J)
78          SUM=Z(J)
79 C
80 C      CALCULATE LAYER DIFFUSIVITY
81          DIFF(J)=XK(J)*PAVG/(POR(J)*VISC)
82 C
83 C      CONVERT DIFFUSIVITY FROM M**2/SEC TO M**2/HR
84          DIFF(J)=3600.0*DIFF(J)
85 C
86          WRITE (*,40000) J
87          READ (*,*) NUM(J)
88          KOUNT=KOUNT+NUM(J)
89      100 CONTINUE
90 C
91      ZBOT=Z(NLayer)
92 C
93 C
94      200 WRITE (*,50000)
95      READ (*,60000) FNAME
96      OPEN (2,FILE=FNAME, STATUS='NEW')
97 C
98      WRITE (*,30000)
99      READ (*,*) F
100     OMEGA=2.*PI*F
101 C
102     WRITE (2,*) KOUNT
103 C
104     CALL HOMOG2(NLayer,OMEGA,AMP,PHASE)
105 C
106     DO 300 J=1,KOUNT
107         WRITE (2,*) ZZ(J),AMP(J),PHASE(J)
108     300 CONTINUE
109 C
110     CLOSE (2)
111     WRITE (*,70000)
112     READ (*,60000) FLAG
113     IF (FLAG.EQ.'Y') GO TO 200
114 C
115     STOP
116 10000 FORMAT(/,' INPUT NO. LAYERS ')
117 20000 FORMAT(/,' INPUT THICKNESS (m). PERMEABILITY (m**2), AND ',/,
118      X ' POROSITY FOR LAYER ', I2)
119 30000 FORMAT(/,' INPUT FREQUENCY (cycles/hr) ')
120 40000 FORMAT(/,' INPUT NO. NODES IN LAYER',I2)
121 50000 FORMAT(/,' ENTER OUTPUT FILENAME')
122 60000 FORMAT(A)
123 70000 FORMAT(/,' ANOTHER FREQUENCY? (Y/N)')
124     END

```

```

125      SUBROUTINE HOMOG2 NLayer,OMEGA,AMP,PHASE)
126 C
127      COMMON /ROCK/XK(10),DIFF(10),Z(10),NUM(10),ZZ(1000)
128 C
129      COMPLEX A(10,10),SQ1,F1,F2,PP,AA(10),BB(10)
130      DIMENSION X(36),B(36,36),AMP(1000),PHASE(1000)
131      DIMENSION C(40,40),LP(40)
132 C
133 C
134 C      HOMOG COMPUTES THE AMPLITUDE AND PHASE RESPONSE OF A
135 C      LAYERED HALF-SPACE DUE TO A SINUSOIDAL PRESSURE VARIATION
136 C      AT THE SURFACE OF A FORM  $P(Z=0)=EXP(IWT)$ , WHERE W IS
137 C      ANG. FR FREQUENCY, T IS TIME, AND I IS SORT(-1).
138 C
139 C      PARAMETERS ARE:
140 C
141 C      NLayer      NUMBER OF LAYERS
142 C      Z(J)        DISTANCE (M) FROM SURFACE TO BOTTOM OF JTH LAYER
143 C                  IE.  $Z(J+1)-Z(J)=H(J+1)$ , WHERE H IS LAYER
144 C                  THICKNESS, J IS LAYER INDEX (J=1 IS TOP LAYER)
145 C      XK(J)       PERMEABILITY (m**2) OF THE JTH LAYER
146 C      DIFF(J)     DIFFUSIVITY (m**2/hr) OF THE JTH LAYER
147 C      OMEGA       ANGULAR FREQUENCY ( $=2\pi f$ )
148 C
149 C
150 C      DIMENSION OF SYSTEM OF EQUATIONS
151      N=2*NLayer-4
152 C
153 C      DIMENSION OF SUB-BLOCK OF SYSTEM
154      NSUB=2*NLayer-2
155 C
156 C      NOTE: COMPLEX COEFFICIENT MATRIX HAS DIMENSION
157 C            NSUB X NSUB
158 C
159 C            REAL ANTI-SYMMETRIC BLOCK MATRIX HAS DIMENSION
160 C            N X N
161 C
162 C      SORT(-1)
163      G1=1.0/SQRT(2.0)
164      SQ1=CMPLX(G1,G1)
165 C
166 C      ZERO OUT ARRAYS
167      DO 200 J=1,NSUB
168          AA(J)=CMPLX(0.0,0.0)
169          BB(J)=CMPLX(0.0,0.0)
170      DO 100 K=1,NSUB
171          A(J,K)=CMPLX(0.0,0.0)
172      100 CONTINUE
173      200 CONTINUE
174 C
175      DO 400 J=1,N
176          X(J)=0.0
177      DO 300 K=1,N
178          B(J,K)=0.0
179      300 CONTINUE
180      400 CONTINUE
181 C
182 C      EVALUATE COMPLEX ELEMENTS OF "A" ARRAY
183      DO 900 J=1,NSUB
184          JM2=J-2
185          JM1=J-1
186          JP1=J+1
187          JP2=J+2
188 C
189 C      IF (MOD(J,2),EQ.0) GO TO 700
190 C

```

```

1 191 C      ODD ROWS OF A
1 192 500      L=(J+1)/2
1 193 C      NOTE: L IS LAYER NUMBER
1 194          LP1=L+1
1 195          ARG1=SQRT(OMEGA/DIFF(L))
1 196          ARG2=SQRT(OMEGA/DIFF(LP1))
1 197 C
1 198          A(J,J) = XK(L)*SQI*ARG1*CEXP(SQI*ARG1*Z(L))
1 199          IF (L.EQ.1) A(J,J)=A(J,J)+XK(L)*SQI*ARG1*CEXP(-SQI*ARG1*Z(L))
1 200          A(J,JP1)= XK(LP1)*SQI*ARG2*CEXP(-SQI*ARG2*Z(L))
1 201          IF (L.EQ.NLAYER) GO TO 600
1 202          A(J,JP2)=-XK(LP1)*SQI*ARG2*CEXP(SQI*ARG2*Z(L))
1 203          IF (L.EQ.1) GO TO 900
1 204 600      A(J,JM1)=-XK(L)*SQI*ARG1*CEXP(-SQI*ARG1*Z(L))
1 205          GO TO 900
1 206 C
1 207 C      EVEN ROWS
1 208 700      L=J/2
1 209 C      NOTE: L IS LAYER NUMBER
1 210          LP1=L+1
1 211 C
1 212          ARG1=SQRT(OMEGA/DIFF(L))
1 213          ARG2=SQRT(OMEGA/DIFF(LP1))
1 214 C
1 215          A(J,JM1)= CEXP(SQI*ARG1*Z(L))
1 216          IF (L.EQ.1) A(J,JM1)=A(J,JM1)-CEXP(-SQI*ARG1*Z(L))
1 217          A(J,J)=- CEXP(-SQI*ARG2*Z(L))
1 218          IF (L.EQ.NLAYER) GO TO 800
1 219          A(J,JP1)=-CEXP(SQI*ARG2*Z(L))
1 220          IF (L.EQ.1) GO TO 900
1 221 800      A(J,JM2)= CEXP(-SQI*ARG1*Z(L))
1 222 900      CONTINUE
1 223 C
1 224 C      BREAK THE COMPLEX NSUB X NSUB MATRIX A INTO THE BLOCK ANTI-
1 225 C      SYMMETRIC REAL N X N MATRIX B, WHERE N=2*NSUB
1 226 C
1 227          DO 1100 J=1,NSUB
1 228          DO 1000 K=1,NSUB
2 229              B(J,K)=REAL(A(J,K))
2 230              B(J+NSUB,K)=AIMAG(A(J,K))
2 231              B(J,K+NSUB)=-AIMAG(A(J,K))
2 232              B(J+NSUB,K+NSUB)=REAL(A(J,K))
2 233 1000      CONTINUE
1 234 1100      CONTINUE
1 235 C
1 236 C
1 237          DO 1300 J=1,N
1 238          DO 1200 K=1,N
2 239              C(J,K)=B(J,K)
2 240 1200      CONTINUE
1 241 1300      CONTINUE
1 242 C
1 243 C      SET UP "RIGHT HAND SIDE" OF REAL MATRIX EQUATION
1 244      ARG=SQRT(OMEGA/DIFF(1))
1 245      F1=XK(1)*SQI*ARG*CEXP(-SQI*ARG*Z(1))
1 246      F2=-CEXP(-SQI*ARG*Z(1))
1 247      X(1)=REAL(F1)
1 248      X(2)=REAL(F2)
1 249      X(NSUB+1)=AIMAG(F1)
1 250      X(NSUB+2)=AIMAG(F2)
1 251 C
1 252      NP1=N+1
1 253 C
1 254          DO 1400 J=1,N
1 255      C(J,NP1)=X(J)
1 256 1400      CONTINUE

```

```

257 C
258 C      SOLVE SET OF SIMULTANEOUS EQUATIONS.
259 C      NOTE: "YENBOB" IS A COPYRIGHTED SUBROUTINE BY
260 C      PEERLESS ENGINEERING SERVICE AND PUBLISHED
261 C      BY JOHN WILEY AND SONS, INC.
262 C
263 C      CALL YENBOB(C,40,N)
264 C
265 C      DO 1500 J=1,N
1 266 C      X(J)=C(J,NP1)
1 267 1500 CONTINUE
268 C
269 C      CALCULATE COMPLEX COEFFICIENTS
270 C      BB(1)=CMPLX(X(1),X(NSUB+1))
271 C      AA(1)=1.000-BB(1)
272 C      DO 1600 J=2,NSUB,2
1 273 C      INDEX=(J/2)+1
1 274 C      AA(INDEX)=CMPLX(X(J),X(NSUB+J))
1 275 C
1 276 C      THE NEXT LINE INSURES THAT BB(NLAYER) IS ZERO.
1 277 C      REQUIRED FOR A FINITE SOLUTION AS Z GETS LARGE
1 278 C      IF (INDEX.EQ.NLAYER) GO TO 1600
1 279 C
1 280 C      BB(INDEX)=CMPLX(X(J+1),X(NSUB+J+1))
1 281 1600 CONTINUE
282 C
283 C      COMPUTE AMPLITUDE(Z) AND PHASE(Z). AMPLITUDE AT SURFACE
284 C      IS 1.0. ALL PHASES ARE WITH RESPECT TO SURFACE SIGNAL.
285 C
286 C      KOUNT=0
287 C      DO 2000 J=1,NLAYER
1 288 C      ARG=SQRT(OMEGA/DIFF(J))
1 289 C      IF(J.EQ.1) GOTO 1700
1 290 C
1 291 C      LAYER THICKNESS
1 292 C      H=Z(J)-Z(J-1)
1 293 C      ZTOP=Z(J-1)
1 294 C      GO TO 1800
1 295 C
1 296 1700 C      H=Z(J)
1 297 C      ZTOP=0.0
1 298 1800 C      DZ=H/FLOAT(NUM(J)-1)
1 299 C
1 300 C      COMPUTE COMPLEX PRESSURE AT EACH DEPTH WITHIN LAYER
1 301 C      J
1 302 C      DO 1900 K=1,NUM(J)
2 303 C      KOUNT=KOUNT+1
2 304 C      ZZ(KOUNT)=FLOAT(K-1)*DZ+ZTOP
2 305 C      PP=AA(J)*CEXP(-SQI*ARG*ZZ(KOUNT))+BB(J)*CEXP(SQI*ARG*ZZ
2 306 C      (KOUNT))
2 307 C      AMP(KOUNT)=CABS(PP)
2 308 C      RAT=AIMAG(PP)/CABS(PP)
2 309 C      PHASE(KOUNT)=ASIN(RAT)
2 310 1900 CONTINUE
1 311 C
1 312 2000 CONTINUE
313 C      RETURN
314 C      END

315 C      SUBROUTINE YENBOB(A,IROW,ICOL)
316 C
317 C      PURPOSE...YENBOB FINDS THE SOLUTION OF A SYSTEM OF N EQUATIONS IN
318 C      N UNKNOWNNS.
319 C
320 C      DIMENSION A(1)

```

```

321      ICOL1 = ICOL + 1
322      DO 800 K = 1, ICOL
1 323      INDEX1 = (K-1)*IROW + K
1 324      A(INDEX1) = 1./A(INDEX1)
1 325      DO 200 J = 1, ICOL1
2 326      IF (J-K) 100, 200, 100
2 327      100      INDEX2 = (J-1)*IROW + K
2 328      A(INDEX2) = A(INDEX2)*A(INDEX1)
2 329      200      CONTINUE
1 330      DO 600 I = 1, ICOL
2 331      IF (I-K) 300, 600, 300
2 332      300      INDEX3 = (K-1)*IROW + I
2 333      DO 500 J = 1, ICOL1
3 334      IF (J-K) 400, 500, 400
3 335      400      INDEX2 = (J-1)*IROW + I
3 336      INDEX4 = (J - 1)*IROW + K
3 337      A(INDEX2) = A(INDEX2) - A(INDEX4)*A(INDEX3)
3 338      500      CONTINUE
2 339      600      CONTINUE
1 340      DO 800 I = 1, ICOL
2 341      IF (I-K) 700, 800, 700
2 342      700      INDEX2 = (K-1)*IROW + I
2 343      A(INDEX2) = -A(INDEX2)*A(INDEX1)
2 344      800      CONTINUE
345      RETURN
346      END

```

COMPUTER PROGRAM: HOMOG3D

HOMOG3D is similar to HOMOG1 and HOMOG1D in that it computes the pressure response of a layered half-space due to either an applied surface step increase or delta increase in pressure. However, it differs in that the output is an arithmetic average pressure over a prescribed depth interval. This interval can span one or more distinct layers. Method of solution is semi-analytical. Calculations are done in double precision.

C)

```

1 C  ++++++
2 C  +//
3 C  +//
4 C  +//
5 C  +//
6 C  +//
7 C  +//
8 C  +//
9 C  +//
10 C  +//
11 C  +//
12 C  +//
13 C  +//
14 C  +//
15 C  +//
16 C  +//
17 C  +//
18 C  +//
19 C  +//
20 C  +//
21 C  +//
22 C  +//
23 C  +//
24 C  +//
25 C  +//
26 C  +//
27 C  +//
28 C  +//
29 C  +//
30 C  +//
31 C  +//
32 C  +//
33 C  +//
34 C  +//
35 C  +//
36 C  +//
37 C  +//
38 C  +//
39 C  +//
40 C  +//
41 C  +//
42 C  +//
43 C  +//
44 C  +//
45 C  +//
46 C  +//
47 C  +//
48 C  +//
49 C  +//
50 C  +//
51 C  +//
52 C  +//
53 C  +//
54 C  +//
55 C  +//
56 C  +//
57 C  +//
58 C  +//
59 C  +//
60 C  +//
61 C  +//
62 C  +//
63 C  +//

```

COMMENT:
 HOMOG3 REPRESENTS ONLY A SLIGHT MODIFICATION TO HOMOG1. THE
 LATTER CALCULATES THE PRESSURE DISTRIBUTION WITH DEPTH FOR A
 GIVEN TIME AND LAYERED HALF-SPACE. HOWEVER, HOMOG3 CALCULATES
 THE AVERAGED PRESSURE RESPONSE OVER A GIVEN DEPTH INTERVAL AS
 A FUNCTION OF TIME. THE DEPTH AVERAGING INTERVAL CAN SPAN ONE
 OR MORE LAYERS IN THE HALF-SPACE. HOWEVER, IF ONE WANTS TO AVERAGE
 OVER A DEPTH INTERVAL WITHIN A LAYER, SIMPLY BREAK THAT LAYER INTO
 SEVERAL LAYERS, EACH HAVING THE SAME FLOW PROPERTIES. ALSO, IF
 AVERAGING OVER SEVERAL LAYERS IS REQUESTED, THOSE LAYERS MUST BE
 ADJACENT LAYERS, AS PRESENTLY CONFIGURED.

THE DRIVING PRESSURE BOUNDARY CONDITION AT THE SURFACE CAN EITHER
 BE A STEP FUNCTION, WHERE $P=0$, TIME $.LT.0$ OR $P=1$, TIME $.GE.0$
 OR A DELTA FUNCTION. THE LATTER WILL BE USEFUL FOR CALCULATING THE
 IMPULSE RESPONSE (IR) OF THE LAYERED HALF-SPACE. CONVOLVING THE IR
 WITH A MEASURED SURFACE BAROMETRIC PRESSURE WILL YIELD A PREDICTED
 RESPONSE AT DEPTH.

A COMPUTER PROGRAM WRITTEN FOR:
 EARTH SCIENCE DEPARTMENT
 UNIVERSITY OF CALIFORNIA
 LAWRENCE LIVERMORE NATIONAL LABORATORY
 LIVERMORE, CALIFORNIA

WRITTEN BY:
 APPLIED GEOPHYSICS GROUP
 TERRA TEK RESEARCH
 420 WAKARA WAY
 SALT LAKE CITY, UTAH 84108

APPLICATION:
 EVALUATION OF THE BAROMETRIC PRESSURE RESPONSE WITHIN A LAYERED
 HALF-SPACE GIVEN A PRESCRIBED SURFACE BAROMETRIC PRESSURE. THE
 SURFACE PRESSURE IS ASSUMED TO HAVE THE FORM:

$$P(t) = L^{\circ}[\text{function}]$$

WHERE L° IS THE NUMERICAL LAPLACE INVERSION OPERATOR AND function
 IS THE SURFACE PRESSURE IN LAPLACE SPACE.

PRESSURE AMPLITUDES ARE CALCULATED AS FUNCTIONS OF DEPTH
 FOR A GIVEN HALF-SPACE CONFIGURATION AND A GIVEN TIME. UP TO
 10 LAYERS OF GIVEN THICKNESS, POROSITY, AND PERMEABILITY CAN BE
 HANDLED (THIS IS EASILY MODIFIED BY CHANGING A FEW DIMENSIONAL
 STATEMENTS). 1-D (VERTICAL) FLOW IS ASSUMED, AND BOTH PRESSURE AND
 FLUX CONTINUITY BOUNDARY CONDITIONS ARE SATISFIED AT EACH INTERFACE.
 ALTHOUGH INPUT AS A FINITE THICKNESS LAYER, THE LOWER BOUNDARY OF
 THE BOTTOM-MOST LAYER REALLY GOES TO INFINITY, WITH A ZERO-PRESSURE
 BOUNDARY CONDITION THERE.

COMMENT: THE NUMBER OF NODES (AN INTERACTIVE REQUEST) DOES NOT EFFECT
 SOLUTION PRECISION. THIS IS SIMPLY THE NUMBER OF LOCATIONS.


```

64 C          THAT THE PRESSURE AMPLITUDE AND PHASE IS TO BE EVALUATED.
65 C
66 C      METHOD:
67 C      HIGH-PRECISION SEMI-ANALYTIC APPROACH USING THE CONCEPT OF
68 C      PROPAGATOR MATRICES.
69 C
70 C *DEBUG
71 C *LARGE
72 C *NOFLOATCALLS
73 C
74 C      PROGRAM DRIVER
75 C      REAL*8 XK(10),DIFF(10),Z(10),V(20),POR(10),H(10)
76 C      REAL*8 PAVG,SUM,TIME,VISC,AMP
77 C      COMMON /ROCK/XK,DIFF,Z,NUM(10),ITYPE,ID(10),NL
78 C      COMMON /LAPL/NLAP,V
79 C
80 C      CHARACTER*20 FNAME,ITYPE
81 C
82 C      SET MS-FORTRAN 8087 MATH ERROR EXCEPTIONS
83 C      (MASK OVERFLOW AND UNDERFLOW)
84 C      CALL LCWRQQ (4927)
85 C
86 C      WRITE (*,80000)
87 C      READ (*,*) NLAP
88 C
89 C      SET UP INVERSE LAPLACE TRANSFORM COEFFICIENT ARRAY
90 C      CALL LINV(NLAP,V)
91 C
92 C      SET VALUES FOR GAS VISCOSITY (pa.sec) AND MEAN GAS PRESSURE (pa)
93 C      VISC=2.0D-05
94 C      PAVG=0.5D04
95 C
96 C      INPUT HALF-SPACE PARAMETERS
97 C      10 WRITE (*,10000)
98 C      READ (*,*) NLAYER
99 C
100 C      KOUNT=0
101 C      SUM=0.0D+00
102 C      DO 100 J=1,NLAYER
103 C          WRITE (*,20000) J
104 C          READ (*,*) A1,A2,A3
105 C          H(J)=DBLE(A1)
106 C          XK(J)=DBLE(A2)
107 C          POR(J)=DBLE(A3)
108 C          SUM=SUM+H(J)
109 C          Z(J)=SUM
110 C
111 C      CALCULATE LAYER DIFFUSIVITY
112 C          DIFF(J)=XK(J)*PAVG/(POR(J)*VISC)
113 C
114 C      CONVERT DIFFUSIVITY FROM M**2/SEC TO M**2/HR
115 C          DIFF(J)=3600.0D+00*DIFF(J)
116 C
117 C      100 CONTINUE
118 C
119 C
120 C      INPUT NO. LAYERS TO AVERAGE AND WHICH LAYERS
121 C      WRITE (*,85000)
122 C      READ (*,*) NL
123 C      DO 150 J=1,NL
124 C          WRITE (*,86000)
125 C          READ (*,*) ID(J),NUM(J)
126 C      150 CONTINUE
127 C
128 C
129 C      WRITE (*,88000)

```

```

130      READ(*,89000) ITYPE
131 C
132      200 WRITE(*,50000)
133      READ(*,60000) FNAME
134      OPEN (2,FILE=FNAME, STATUS='NEW')
135 C
136 C      INPUT NO. TIME POINTS AND DELTA TIME
137      WRITE(*,87000)
138      READ(*,*) NT,DT,TSTART
139      WRITE(2,*) NT
140 C
141      DO 300 J=1,NT
142      TTT=FLOAT(J-1)*DT+TSTART
143      TIME=DBLE(TTT)
144 C
145      CALL HOMOG1(NLAYER,TIME,AMP)
146 C
147      A1=SNGL(AMP)
148      WRITE (2,*)TTT,A1
149      WRITE(*,88500) TTT,A1
150      300 CONTINUE
151 C
152      CLOSE (2)
153      WRITE (*,70000)
154      READ (*, 60000) FLAG
155      IF (FLAG.EQ.'Y') GO TO 10
156 C
157      STOP
158 10000 FORMAT(/,' INPUT NO. LAYERS ')
159 20000 FORMAT(/,' INPUT THICKNESS (m), PERMEABILITY (m**2), AND ',/,
160 X ' POROSITY FOR LAYER ', I2)
161 50000 FORMAT(/,' ENTER OUTPUT FILENAME')
162 60000 FORMAT(A)
163 70000 FORMAT(/, ' ANOTHER HALF-SPACE? (Y/N)')
164 80000 FORMAT(/, ' INPUT NLAP ((=16)')
165 85000 FORMAT(/, ' INPUT NO LAYERS TO AVERAGE OVER')
166 86000 FORMAT(/, ' INPUT LAYER NUMBER AND NO. NODES IN LAYER')
167 87000 FORMAT(/, ' INPUT NT,DT, AND TSTART')
168 88000 FORMAT(/, ' IMPULSE (I) OR STEP (S)')
169 88500 FORMAT(/, ' TIME (hrs) =',E12.6,2X,' PRESSURE =',E12.6)
170 89000 FORMAT(A)
171      END

172      SUBROUTINE HOMOG1(NLAYER,TIME,ANS)
173 C
174      REAL*8 XK(10),DIFF(10),Z(10),V(20),AMP2(1000,20)
175      REAL*8 A(10,10),AA(10,20),BB(10,20),X(10),C(20,20)
176      REAL*8 AMP(1000)
177      REAL*8 AB,ARG,ARG1,ARG2,DZ,F1,F2,FUN,H,PP,S
178      REAL*8 SUM,TIME,ZTOP,SUM2,ANS,THICK,ZZ,FAC
179 C
180      CHARACTER*20 ITYPE
181      COMMON /ROCK/XK,DIFF,Z,NUM(10),ITYPE,ID(10),NL
182      COMMON /LAPL/NLAP,V
183 C
184 C
185 C
186 C      HOMOG1 COMPUTES THE PRESSURE AMPLITUDE RESPONSE OF A
187 C      LAYERED HALF-SPACE DUE TO A PRESCRIBED PRESSURE VARIATION
188 C      AT THE SURFACE OF A FORM:
189 C
190 C      P(t) = L'[function]
191 C
192 C      WHERE L' IS THE NUMERICAL LAPLACE INVERSE OPERATOR AND
193 C      function IS THE SURFACE PRESSURE IN LAPLACE SPACE.

```

```

194 C
195 C     PARAMETERS ARE:
196 C
197 C     NLAYER  NUMBER OF LAYERS
198 C     Z(J)    DISTANCE (M) FROM SURFACE TO BOTTOM OF JTH LAYER
199 C             IE.  $Z(J+1)-Z(J) = H(J+1)$ , WHERE H IS LAYER
200 C             THICKNESS, J IS LAYER INDEX (J=1 IS TOP LAYER)
201 C     XK(J)    PERMEABILITY ( $m^2/s$ ) OF THE JTH LAYER
202 C     DIFF(J)  DIFFUSIVITY ( $m^2/s$ ) OF THE JTH LAYER
203 C
204 C
205 C     DIMENSION OF SYSTEM OF EQUATIONS
206 C     N=2*NLAYER-2
207 C
208 C     IF (TIME.GT.0.0D0) GOTO 5
209 C     ANS=0.0D0
210 C     GOTO 250D
211 C     S=DLG(2.0D+00)/TIME
212 C
213 C     DO 1300 KK=1,NLAP
214 C     S=DBLE(FLOAT(KK))*AB
1 215 C
1 216 C     PRESCRIBE LAPLACE TRANSFORM OF SURFACE PRESSURE
1 217 C     IF (ITYPE.EQ.'S') FUN=1.0D+00/S
1 218 C     IF (ITYPE.EQ.'I') FUN=1.0D0
1 219 C
1 220 C     ZERO OUT ARRAYS
1 221 C     DO 200 J=1,N
2 222 C     X(J)=0.0D+00
2 223 C     AA(J,KK)=0.0D+00
2 224 C     BB(J,KK)=0.0D+00
2 225 C     DO 100 K=1,N
3 226 C     A(J,K)=0.0D+00
3 227 C     100 CONTINUE
2 228 C     200 CONTINUE
1 229 C
1 230 C     EVALUATE ELEMENTS OF "A" ARRAY
1 231 C     DO 700 J=1,N
2 232 C     JM2=J-2
2 233 C     JM1=J-1
2 234 C     JP1=J+1
2 235 C     JP2=J+2
2 236 C
2 237 C     IF (MOD(J,2).EQ.0) GO TO 500
2 238 C
2 239 C     ODD ROWS OF A
2 240 C     300 L=(J+1)/2
2 241 C     NOTE: L IS LAYER NUMBER
2 242 C     LP1=L+1
2 243 C     ARG1=DSQRT(S/DIFF(L))
2 244 C     ARG2=DSQRT(S/DIFF(LP1))
2 245 C
2 246 C     A(J,J) = XK(L)*ARG1*DEXP(ARG1*Z(L))
2 247 C     IF (L.EQ.1) A(J,J)=A(J,J)+XK(L)*ARG1*DEXP(-ARG1*Z(L))
2 248 C     A(J,JP1)=XK(LP1)*ARG2*DEXP(-ARG2*Z(L))
2 249 C     IF (L.EQ.NLAYER) GO TO 400
2 250 C     A(J,JP2)=-XK(LP1)*ARG2*DEXP(ARG2*Z(L))
2 251 C     IF (L.EQ.1) GO TO 700
2 252 C     400 A(J,JM1)=-XK(L)*ARG1*DEXP(-ARG1*Z(L))
2 253 C     GO TO 700
2 254 C
2 255 C     EVEN ROWS
2 256 C     500 L=J/2
2 257 C     NOTE: L IS LAYER NUMBER
2 258 C     LP1=L+1
2 259 C

```

```

2 260 ARG1=DSQRT(S/DIFF(L))
2 261 ARG2=DSQRT(S/DIFF(LP1))
2 262 C
2 263 A(J,JM1)=DEXP(ARG1*Z(L))
2 264 IF (L.EQ.1) A(J,JM1)=A(J,JM1)-DEXP(-ARG1*Z(L))
2 265 A(J,J)=-DEXP(-ARG2*Z(L))
2 266 IF (L.EQ.NLAYER) GO TO 600
2 267 A(J,JP1)=-DEXP(ARG2*Z(L))
2 268 IF (L.EQ.1) GO TO 700
2 269 600 A(J,JM2)=DEXP(-ARG1*Z(L))
2 270 700 CONTINUE
1 271 C
1 272 C
1 273 DO 900 J=1,N
2 274 DO 800 K=1,N
3 275 C(J,K)=A(J,K)
3 276 800 CONTINUE
2 277 900 CONTINUE
1 278 C
1 279 C SET UP "RIGHT HAND SIDE" OF REAL MATRIX EQUATION
1 280 ARG=DSQRT(S/DIFF(1))
1 281 F1=XK(1)*ARG*DEXP(-ARG*Z(1))
1 282 F2=-DEXP(-ARG*Z(1))
1 283 X(1)=FUN*F1
1 284 X(2)=FUN*F2
1 285 C
1 286 NP1=N+1
1 287 C
1 288 DO 1000 J=1,N
2 289 C(J,NP1)=X(J)
2 290 1000 CONTINUE
1 291 C
1 292 C
1 293 C SOLVE SET OF SIMULTANEOUS EQUATIONS
1 294 C NOTE: "YENBOB" IS A COPYRIGHTED SUBROUTINE BY
1 295 C PEARLESS ENGINEERING SERVICE FOR
1 296 C JOHN WILEY AND SONS, INC.
1 297 C
1 298 CALL YENBOB(C,Z0,N)
1 299 C
1 300 DO 1100 J=1,N
2 301 X(J)=C(J,NP1)
2 302 1100 CONTINUE
1 303 C
1 304 C CALCULATE LAYER COEFFICIENTS AA AND BB
1 305 BB(1,KK)=X(1)
1 306 AA(1,KK)=FUN-BB(1,KK)
1 307 DO 1200 J=2,N,2
2 308 INDEX=(J/2)+1
2 309 AA(INDEX,KK)=X(J)
2 310 C
2 311 C THE NEXT LINE INSURES THAT BB(NLAYER) IS ZERO,
2 312 C REQUIRED FOR A FINITE SOLUTION AS Z GETS LARGE
2 313 IF (INDEX.EQ.NLAYER) GO TO 1200
2 314 C
2 315 BB(INDEX,KK)=X(J+1)
2 316 1200 CONTINUE
1 317 1300 CONTINUE
1 318 C
1 319 C COMPUTE AMPLITUDE(Z).
1 320 C
1 321 DO 1400 KK=1,NLAP
1 322 S=DSQLE(FLOAT(KK))*AB
1 323 C
1 324 KOUNT=0
1 325 DO 1700 L=1,NL

```

```

2 326 J=ID(L)
2 327 ARG=DSORT(S/DIFF(J))
2 328 IF (J.EQ.1) GO TO 1400
2 329 C
2 330 C LAYER THICKNESS
2 331 H=Z(J)-Z(J-1)
2 332 ZTOP=Z(J-1)
2 333 GO TO 1500
2 334 C
2 335 1400 H=Z(J)
2 336 ZTOP=0.0D+00
2 337 1500 DZ=H/DBLE(FLOAT(NUM(L)-1))
2 338 C
2 339 C COMPUTE PRESSURE AT EACH DEPTH WITHIN LAYER
2 340 C J
2 341 DO 1600 K=1, NUM(L)
2 342 KOUNT=KOUNT+1
2 343 ZZ=DBLE(FLOAT(K-1))*DZ+ZTOP
2 344 PP=AA(J, KK)*DEXP(-ARG*ZZ)+BB(J, KK)*DEXP(ARG*ZZ)
2 345 AMP2(KOUNT, KK)=PP
2 346 1600 CONTINUE
2 347 1700 CONTINUE
1 348 1800 CONTINUE
2 349 C
2 350 C COMPUTE INVERSE LAPLACE TRANSFORM AT EACH DEPTH
2 351 DO 2000 J=1, KOUNT
1 352 SUM=0.0D+00
1 353 DO 1900 KK=1, NLAP
2 354 SUM=SUM+V(KK)*AMP2(J, KK)
2 355 1900 CONTINUE
1 356 AMP(J)=AB*SUM
1 357 2000 CONTINUE
2 358 C
2 359 C CALCULATE AVERAGE PRESSURE
2 360 C USE TRAPEZOIDAL RULE TO INTEGRATE OVER EACH LAYER
2 361 C TOTAL ALLOWED NO INTEGRATION POINTS IS 1000
2 362 KOUNT=0
2 363 SUM2=0.0D0
2 364 THICK=0.0D0
2 365 DO 2300 I=1, NL
1 366 J=ID(L)
1 367 IF (J.EQ.1) H=Z(J)
1 368 IF (J.GT.1) H=Z(J)-Z(J-1)
1 369 DZ=H/DBLE(FLOAT(NUM(L)-1))
1 370 THICK=THICK+H
1 371 DO 2200 K=1, NUM(L)
2 372 FAC=1.0D0
2 373 IF (K.EQ.1) FAC=0.5D0
2 374 IF (K.EQ.NUM(L)) FAC=0.5D0
2 375 KOUNT=KOUNT+1
2 376 SUM2=SUM2+FAC*AMP(KOUNT)*DZ
2 377 2200 CONTINUE
1 378 2300 CONTINUE
2 379 C
2 380 ANS=SUM2/THICK
2 381 2500 RETURN
2 382 END

383 SUBROUTINE YENBOB(A, IROW, ICOL)
384 C
385 C PURPOSE...YENBOB FINDS THE SOLUTION OF A SYSTEM OF N EQUATIONS IN
386 C N UNKNOWNNS.
387 C
388 REAL*8 A(1000)
389 ICOL1 = ICOL + 1

```

```

390      DO 800 K = 1, ICOL
1 391      INDEX1 = (K-1)*IROW + K
1 392      A(INDEX1) = 1.D+00/A(INDEX1)
1 393      DO 200 J = 1, ICOL1
2 394      IF (J-K) 100, 200, 100
2 395      100      INDEX2 = (J-1)*IROW + K
2 396      A(INDEX2) = A(INDEX2)*A(INDEX1)
2 397      200      CONTINUE
1 398      DO 600 I = 1, ICOL
2 399      IF (I-K) 300, 600, 300
2 400      300      INDEX3 = (K-1)*IROW + I
2 401      DO 500 J = 1, ICOL1
3 402      IF (J-K) 400, 500, 400
3 403      400      INDEX2 = (J-1)*IROW + I
3 404      INDEX4 = (J - 1)*IROW + K
3 405      A(INDEX2) = A(INDEX2) - A(INDEX4)*A(INDEX3)
3 406      500      CONTINUE
2 407      600      CONTINUE
1 408      DO 800 I = 1, ICOL
2 409      IF (I-K) 700, 800, 700
2 410      700      INDEX2 = (K-1)*IROW + I
2 411      A(INDEX2) = -A(INDEX2)*A(INDEX1)
2 412      800      CONTINUE
413      RETURN
414      END

415      SUBROUTINE LINV(N,V)
416      REAL*8 V(20), G(100), H(100), SN
417      C
418      G(1)=1.0D+00
419      NH=N/2
420      DO 100 I=1, N
1 421      G(I+1)=G(I)*DBLE(FLOAT(I))
1 422      100      CONTINUE
423      H(1)=2.0D+00/G(NH)
424      DO 200 I=2, NH
1 425      H(I)=(DBLE(FLOAT(I))*NH)*G(2*I+1)/(G(NH-I+1)*G(I+1)+G(I))
1 426      200      CONTINUE
427      SN=2.0D+00*DBLE(FLOAT(MOD(NH, 2)))-1.0D+00
428      DO 400 I=1, N
1 429      V(I)=0.0D+00
1 430      NTOP=MIN0(I, NH)
1 431      NBOT=(I+1)/2
1 432      DO 300 K=NBOT, NTOP
2 433      V(I)=V(I)+H(K)/(G(I-K+1)*G(2*K-I+1))
2 434      300      CONTINUE
1 435      V(I)=SN*V(I)
1 436      SN=-SN
1 437      400      CONTINUE
438      RETURN
439      END

```

COMPUTER PROGRAM: RUNBFD

RUNBFD sets up the required input data files for computer programs BFD and BFD2.

```

D)
1 C      ++++++
2 C      +//////////
3 C      +/////  RUNBFD (Run Barometric Finite Difference program)  +/////
4 C      +//////////
5 C      ++++++
6 C
7 C      A COMPUTER PROGRAM WRITTEN FOR:
8 C      EARTH SCIENCES DEPARTMENT
9 C      UNIVERSITY OF CALIFORNIA
10 C     LAWRENCE LIVERMORE NATIONAL LABORATORY
11 C     LIVERMORE, CALIFORNIA
12 C
13 C     WRITTEN BY:
14 C     APPLIED GEOPHYSICS GROUP
15 C     TERRA TEK RESEARCH
16 C     420 WAKARA WAY
17 C     SALT LAKE CITY, UTAH 84108
18 C     OCTOBER, 1984
19 C
20 C     APPLICATION:
21 C     RUNBFD SETS UP REQUIRED DATA FILES FOR INPUT INTO THE
22 C     PROGRAM "BFD"
23 C
24 C     HARDWARE REQUIREMENTS:
25 C     IBM-PC OR IBM-XT W/
26 C         (A) 8087 COPROCESSOR
27 C         (B) 640K MEMORY
28 C         (C) IF NOT AN XT, A HARD DISK EXPANSION UNIT IS RECOMMENDED
29 C         (D) DOT MATRIX PRINTER (FOR FILE OUTPUT)
30 C
31 C
32 C
33 C *DEBUG
34 C *STORAGE=2
35 C *NOFLOATCALLS
36 C *LARGE P,SF,C,Q
37 C     DIMENSION C(20,200,2),SF(20,200),P(20,200),Q(20,200),
38 C     X  RC(20),ZC(200),PERM(10),POR(10),THICKL(10),NV(10),NBC(10)
39 C     CHARACTER*20 OUTPUT
40 C     PI=3.141592654
41 C     TWOPI=2.0*PI
42 C
43 C     SET MS-FORTRAN 8087 MATH ERROR EXCEPTIONS
44 C     (MASK OVERFLOW AND UNDERFLOW)
45 C     CALL LCWRDQ (4927)
46 C
47 C     ENTER HOLE-FORMATION GEOMETRY, AND FORMATION FLOW PARAMETERS
48 C
49 C     100 WRITE (*,10000)
50 C     READ (*,15000,ERR=100) OUTPUT
51 C     OPEN (2,FILE=OUTPUT,STATUS='NEW')
52 C     200 WRITE (*,20000)
53 C     READ (*,*,ERR=200) VIS,PAVG
54 C     300 WRITE (*,25000)
55 C     READ (*,*,ERR=300) RMIN,RMAX
56 C     400 WRITE (*,30000)
57 C     READ (*,*,ERR=400) NLAYER
58 C     DO 800 N=1,NLAYER
59 C     500     WRITE (*,35000) N
60 C     WRITE (*,40000)
61 C     READ (*,*,ERR=500) PERM(N),POR(N),THICKL(N)
62 C     600     WRITE (*,45000)
63 C     READ (*,*,ERR=600) NV(N)
64 C     IF (NV(N) .LT. 0) GO TO 600

```



```

65 700 WRITE (*,50000)
66 READ (*,*,ERR=700) NBC(N)
67 800 CONTINUE
68 C
69 C COMPUTE NODAL COORDINATES
70 WRITE (*,74000)
71 C
72 C
73 RC(1)=0.0
74 I=1
75 900 I=I+1
76 RC(I)=RMIN*2.0**(I-1)
77 IF (RC(I)-RMAX) 900,1000,1000
78 1000 RC(I)=RMAX
79 NC=I
80 IF (NC.GT.20) GOTO 71000
81 D=0
82 ZC(1)=0
83 NR=1
84 DO 1200 I=1,NLAYER
85 DD=THICKL(I)/NV(I)
86 DO 1100 J=1,NV(I)
87 D=D+DD
88 NR=NR+1
89 1100 ZC(NR)=D
90 1200 NV(I)=NR
91 C
92 IF (NR.GT.200) GOTO 73000
93 C
94 C COMPUTE NODAL CONDUCTIVITIES AND STORAGE FACTORS
95 C
96 L=1
97 DO 1400 J=1,NR
98 IF (J.GT.NV(L)) L=L+1
99 CL=PERM(L)/VIS
100 SFL=POR(L)/PAVG
101 DZ=ZC(MIN0(J+1,NR))-ZC(MAX0(J-1,1))
102 DO 1300 I=1,NC
103 DR=RC(MIN0(I+1,NC))-RC(MAX0(I-1,1))
104 IF (I.NE.NC) C(I,J,2)=CL*DZ/((RC(I+1)-RC(I))*2.0)
105 IF (J.NE.NR) C(I,J,1)=CL*DR/((ZC(J+1)-ZC(J))*2.0)
106 SF(I,J)=SFL*DZ*DR/4.0
107 P(I,J)=PAVG
108 1300 Q(I,J)=0.0
109 C
110 C SET BOUNDARY NODES
111 C
112 IF (NBC(L).EQ.2) GO TO 1400
113 SF(1,J)=SF(1,J)/POR(L)
114 C(1,J,2)=NBC(L)*2.0*C(1,J,2)+0.75/0.5
115 C(1,J,1)=C(1,J,1)*(RMIN**2.0)/(0.0*PERM(L))
116 1400 C(NC,J,2)=0.0
117 DO 1500 I=1,NC
118 1500 C(I,NR,1)=0.0
119 C
120 C REFINE NODAL VALUES ON BOUNDARIES OF HORIZONTAL LAYERS
121 C
122 IF (NLAYER.EQ.1) GO TO 1700
123 DO 1600 L=1,NLAYER-1
124 DO 1600 I=1,NC
125 C(I,NV(L),1)=C(I,NV(L)+1,1)
126 C(I,NV(L),2)=(C(I,NV(L)-1,2)+C(I,NV(L)+1,2))/2.0
127 1600 SF(I,NV(L))=(SF(I,NV(L)-1)+SF(I,NV(L)+1))/2.0
128 C
129 C CONVERT FOR RADIAL FLOW WIDTHS
130 C

```

```

131 1700 DO 1900 J=1,NR
132      DO 1800 I=2,NC-1
133          C(I,J,1)=C(I,J,1)*(RC(I-1)+RC(I+1))*PI
134          SF(I,J)=SF(I,J)*(RC(I-1)+RC(I+1))*PI
135 1800 C(I,J,2)=C(I,J,2)*(RC(I)+RC(I+1))*PI
136      C(NC,J,1)=C(NC,J,1)*(RC(NC-1)+0.75*(RC(NC)-RC(NC-1)))*TWOPI
137      SF(NC,J)=SF(NC,J)*(RC(NC-1)+0.75*(RC(NC)-RC(NC-1)))*TWOPI
138      SF(1,J)=SF(1,J)*RC(2)*PI/2.0
139      C(1,J,1)=C(1,J,1)*0.25*RC(2)*PI
140 1900 C(1,J,2)=C(1,J,2)*0.50*RC(2)*PI
141 C
142 C      WRITE INPUT FILE
143 C
144      WRITE (2,55000) VIS,PAVG,NLAYER,NC,NR
145      WRITE (2,60000) (RC(I),I=1,NC)
146      WRITE (2,60000) (ZC(I),I=1,NR)
147      WRITE (2,65000) (NV(I),NBC(I),PERM(I),POR(I),I=1,NLAYER)
148      DO 2000 J=1,NR
149          DO 2000 I=1,NC
150              WRITE (2,60000) C(I,J,1),C(I,J,2),SF(I,J),P(I,J),Q(I,J)
151 2000      CONTINUE
152      GOTO 75000
153      CLOSE (2)
154 10000 FORMAT(' ENTER OUTPUT FILE NAME:')
155 15000 FORMAT(A)
156 20000 FORMAT(/,' ENTER FLUID VISCOSITY (Pa-sec) AND',
157      X' MEAN PRESSURE (Pa):')
158 25000 FORMAT(/,' ENTER WELL RADIUS (m) AND OUTER RADIUS (m):')
159 30000 FORMAT(/,' ENTER NUMBER OF LAYERS:')
160 35000 FORMAT(///,' FOR LAYER NUMBER:',I3)
161 40000 FORMAT(/,' ENTER PERMEABILITY (sq.m), POROSITY, AND'
162      X' THICKNESS (m):')
163 45000 FORMAT(/,' ENTER NUMBER OF VERTICAL NODES IN LAYER',
164      X' MINIMUM OF 2):')
165 50000 FORMAT(/,' ENTER LAYER BOUNDARY CONDITION AT WELL BORE',/,
166      X' 0=CASED, 1=OPEN HOLE, 2=NO HOLE')
167 55000 FORMAT(2E15.6,3I15)
168 60000 FORMAT(5E15.6)
169 65000 FORMAT(2I15,2E15.6)
170 70000 FORMAT(' OOPS..NEED MORE THAN 20 RADIAL NODES',/,
171      X' CHANGE DIMENSION STATEMENTS OF BFD AND RUNBFD')
172 71000 WRITE(*,70000)
173      GOTO 75000
174 72000 FORMAT(' OOPS..NEED MORE THAN 200 VERTICAL NODES',/,
175      X' CHANGE DIMENSION STATEMENTS OF BFD AND RUNBFD',/,
176      X' OR REDUCE NO. NODES ASSIGNED TO LAYERS')
177 73000 WRITE(*,72000)
178 74000 FORMAT(/,' WRITING FINITE DIFFERENCE MESH DATA FILE')
179 75000 STOP
180      END

```

COMPUTER PROGRAM: BFD
BFD2

BFD and BFD2 compute the pressure response of a partially cased hole intersecting a layered half-space. The top of the casing is sealed to the atmosphere. As presently set up, BFD computes the pressure response to a step increase in pressure at the surface. Method of solution is numerical (finite difference).

C)

```

1 C  ++++++
2 C  +//////////
3 C  +////////// BFD (Barometric Finite Difference program) +//////////
4 C  +//////////
5 C  ++++++
6 C
7 C  A COMPUTER PROGRAM WRITTEN FOR:
8 C  EARTH SCIENCE DEPARTMENT
9 C  UNIVERSITY OF CALIFORNIA
10 C  LAWRENCE LIVERMORE NATIONAL LABORATORY
11 C  LIVERMORE, CALIFORNIA
12 C
13 C  WRITTEN BY:
14 C  APPLIED GEOPHYSICS GROUP
15 C  TERRA TEK RESEARCH
16 C  420 WAKARA WAY
17 C  SALT LAKE CITY, UTAH 84108
18 C  OCTOBER, 1984
19 C
20 C  APPLICATION:
21 C  EVALUATION OF THE BAROMETRIC PRESSURE RESPONSE INSIDE A VERTICAL
22 C  CYLINDRICAL HOLE PENETRATING A LAYERED HALF-SPACE. THE HOLE IS
23 C  CASED FROM THE SURFACE TO SOME PRESCRIBED DEPTH. FROM THIS
24 C  DEPTH, THE HOLE CAN BE (1) OPEN, (2) CASED, OR (3) NON-EXISTANT.
25 C  THE HOLE IS SEALED FROM THE ATMOSPHERE AT THE SURFACE. EACH
26 C  LAYER IS GIVEN A PRESCRIBED PERMEABILITY AND POROSITY.
27 C  HOMOGENIETY AND ISOTROPY ARE ASSUMED IN THE LATERAL PLANE.
28 C  2-D FLOW (RADIAL AND VERTICAL) ASSUMED. A NO-FLOW BOUNDARY IS
29 C  ASSUMED AT THE BOTTOM AND OUTER BOUNDARY OF THE MESH.
30 C
31 C  CURRENT CONFIGURATION:
32 C  AS PRESENTLY CONFIGURED, BFD CAN DEAL WITH UP TO 10 LAYERS, EACH
33 C  OF WHICH CAN HAVE DIFFERENT THICKNESSES, NUMBER OF VERTICAL NODE
34 C  AND FLOW PROPERTIES. A MAXIMUM OF 200 NODES VERTICALLY AND 20
35 C  NODES RADIALY ARE IMPOSED. RADIAL NODES ARE SPACED IN A
36 C  GEOMETRICALLY INCREASING DISTANCE FROM THE WELLBORE. EG. THE
37 C  FIRST NODE (FROM THE HOLE FACE) IS TWICE THE HOLE RADIUS. THE
38 C  SECOND NODE IS FOUR TIMES THE HOLE RADIUS, AND SO ON. NUMBER OF
39 C  RADIAL NODES IS GOVERNED BY THE USER-SUPPLIED HOLE RADIUS AND
40 C  OUTER "NO-FLOW" BOUNDARY.
41 C
42 C  HARDWARE REQUIREMENTS
43 C  IBM-PC OR IBM-XT W/
44 C  (A) 8087 COPROCESSOR
45 C  (B) 640K MEMORY
46 C  (C) IF NOT AN XT, A HARD DISK EXPANSION UNIT IS RECOMMENDED
47 C  (D) DOT MATRIX PRINTER (FOR FILE OUTPUT)
48 C
49 C  BASED ON:
50 C  PRICKETT, T.A. AND C.G. LONNGQUIST
51 C  SELECTED DIGITAL COMPUTER TECHNIQUES FOR GROUNDWATER
52 C  RESOURCE EVALUATION
53 C  ILLINOIS STATE WATER SURVEY, URBANA, BULLETIN 55, 1971
54 C
55 C  DEFINITION OF VARIABLES
56 C
57 C  NSTEPS - NUMBER OF TIME STEPS
58 C  DELTA - SIZE OF TIME STEP (Secs)
59 C  ERROR - MAXIMUM ALLOWED CONVERGENCE ERROR (Pa)
60 C  NC - NUMBER OF NODAL COLUMNS
61 C  NR - NUMBER OF NODAL ROWS
62 C  P(I,J) - NODAL PRESSURE AT CURRENT TIME STEP (Pa)

```

```

63 C      PD(I,J) - NODAL PRESSURE AT PREVIOUS TIME STEP (Pa)
64 C      DL(I,J) - CHANGE IN PRESSURE FROM PREVIOUS TIME STEP (Pa)
65 C      SF(I,J) - STORAGE COEFFICIENT (POROSITY * NODAL AREA / MEAN
66 C      ABSOLUTE PRESSURE) (cc.m./Pa)
67 C      Q(I,J) - TRUE VOLUMETRIC FLUX PER UNIT THICKNESS (cu.m./sec)
68 C      QT - TOTAL VOLUME INFLUX INTO WELL (cu.m)
69 C      C(1,J,1) - NODAL CONDUCTIVITY IN Y-DIRECTION (sq. meter/Pa-sec)
70 C      C(1,J,2) - NODAL CONDUCTIVITY IN X-DIRECTION
71 C      PAVG - AVERAGE ATMOSPHERIC PRESSURE (Pa)
72 C      VIS - VISCOSITY OF FLUID (Pa-sec)
73 C      PERM(I) - PERMEABILITY OF LAYER I (sq.m.)
74 C      POR(I) - POROSITY OF LAYER I
75 C
76 C      *DEBUG
77 C      *STORAGE:2
78 C      *NOFLOATCALLS
79 C      *LARGE P,P0,SF,C,DL,Q
80 C      DIMENSION P(20,200),PD(20,200),SF(20,200),
81 C      X C(20,200,2),DL(20,200),RC(20),ZC(200),Q(20,200),
82 C      X NV(10),NBC(10),PERM(10),POR(10),COMP(10)
83 C      DOUBLE PRECISION B(200),G(200),W,AA,BB,CC,DD
84 C      CHARACTER*20 INPUT,OUTPUT,PINPUT,FNAME,GNAME,HNAME,COMP
85 C      CHARACTER*20 RNAME
86 C
87 C      SET MS-FORTRAN 8007 MATH ERROR EXCEPTIONS
88 C      (MASK OVERFLOW AND UNDERFLOW)
89 C      CALL LCWRQD (4927)
90 C
91 C
92 C      READ INPUT PARAMETERS
93 C
94 C      WRITE (*,10000)
95 C      READ (*,16000) INPUT
96 C      OPEN (1,FILE=INPUT,STATUS='OLD')
97 C      WRITE (*,18000)
98 C      READ (*,16000) OUTPUT
99 C      OPEN (2,FILE=OUTPUT,STATUS='NEW')
100 C      WRITE (*,20000)
101 C      READ (*,*) NSTEPS,IOUT
102 C      WRITE (*,22000)
103 C      READ (*,*) DELTA,ERROR
104 C      CHANGE DELTA TO SECONDS
105 C      DELTA=DELTA*3600.
106 C      WRITE (*,24000)
107 C      READ (*,*) MODE
108 C
109 C
110 C      SET UP PRESSURE VS TIME OUTPUT FILE
111 C      WRITE (*,30000)
112 C      READ (*,16000) FNAME
113 C      OPEN (3,FILE=FNAME,STATUS='NEW')
114 C
115 C      SET UP OUTPUT DATA FILES OF PRESSURE VS DEPTH AT WELLBORE.
116 C      PRESSURE VS DEPTH AT OUTER BOUNDARY, AND DP/DR AT WELLBORE.
117 C      AT FINAL TIME ITERATION
118 C      WRITE (*,52000)
119 C      READ (*,16000) GNAME
120 C      WRITE (*,54000)
121 C      READ (*,16000) HNAME
122 C      WRITE (*,55000)
123 C      READ (*,16000) RNAME
124 C
125 C      OPEN (4,FILE=GNAME, STATUS='NEW')
126 C      OPEN (5,FILE=HNAME, STATUS='NEW')
127 C      OPEN (7,FILE=RNAME, STATUS='NEW')
128 C

```

```

129 C
130 C   READ INFORMATION FROM DISK FILE (GENERATED BY RUNBFD)
131 C
132 C
133     READ (1,26000) VIS,PAVG,NLAYER,NC,NR
134     READ (1,28000) (RC(I),I=1,NC)
135     READ (1,28000) (ZC(I),I=1,NR)
136     READ (1,30000) (NV(I),NBC(I),PERM(I),POR(I),I=1,NLAYER)
137 C
138 C       SET CONSTANT PRESSURE BOUNDARY (SURFACE)
139 C
140   100 WRITE (*,32000)
141     READ (*,*,ERR=100) NTC
142     IF (NTC.EQ.2) GO TO 300
143   200 WRITE (*,34000)
144     READ (*,*,ERR=200) PUPPER
145     GO TO 400
146 C
147   300 WRITE (*,10000)
148     READ (*,16000,ERR=300) PINPUT
149     OPEN (5,FILE=PINPUT,STATUS='OLD')
150 C
151   400     DO 500 J=1,NLAYER
152           IF (NBC(J).EQ.0) COMP(J)='CASED'
153           IF (NBC(J).EQ.1) COMP(J)='OPEN HOLE'
154           IF (NBC(J).EQ.2) COMP(J)='NO HOLE'
155   500     CONTINUE
156 C
157 C       READ NODE CARDS
158 C
159     WRITE (*,12000)
160     DO 600 J=1,NR
161           DO 600 I=1,NC
162           READ (1,28000) C(I,J,1),C(I,J,2),SF(I,J),P(I,J),D(I,J)
163           IF (MODE.EQ.1) P(I,J)=P(I,J)-PAVG
164           PD(I,J)=P(I,J)
165   600     CONTINUE
166 C
167     REWIND(1)
168     CLOSE(1)
169 C
170 C       SET UPPER BOUNDARY
171     IF (NTC.EQ.2) GO TO 300
172   700     DO 800 I=2,NC
173           SF(I,1)=1.0E25
174           P(I,1)=PUPPER
175   800     PD(I,1)=PUPPER
176     GO TO 1100
177   900     DO 1000 I=2,NC
178   1000     SF(I,1)=1.0E25
179     READ (6,56000) PUPPER,ETIME
180     GO TO 700
181 C
182 C       PRINT INPUT PARAMETERS
183 C
184   1100 WRITE (2,36000) DELTA,ERROR,VIS,PAVG
185     WRITE (2,38000)
186     WRITE (2,40000) (I,PERM(I),POR(I),COMP(I),I=1,NLAYER)
187     WRITE (2,42000)
188     WRITE (2,44000) (I,RC(I),I=1,NC)
189     WRITE (2,46000)
190     L=1
191     DO 1200 J=1,NR
192     IF (J.GT.NV(L)) L=L+1
193     WRITE (2,48000) J,L,ZC(J)
194   1200 CONTINUE

```

```

195 C
196 C
197 C      INITIALIZE OUTPUT DATA FILES
198      WRITE (3,*) NSTEPS
199      WRITE (4,*) NR
200      WRITE (5,*) NR
201      WRITE (7,*) NR
202 C
203 C
204 C      START OF SIMULATION
205 C
206      WRITE (*,14000)
207 1300 TIME=0.0
208      QT=0.0
209      DO 4500 ISTEP=1,NSTEPS
210          TIME=TIME+DELTA
211 C
212 C      READ UPPER BOUNDARY PRESSURES IF TIME VARYING
213 C
214      IF (NTC.EQ.1) GO TO 1500
215      IF (TIME.LT.ETIME) GO TO 1500
216      READ (6,56000) PUPPER,ETIME
217      DO 1400 I=2,NC
218 1400      P(I,1)=PUPPER
219 C
220 C      PREDICT PRESSURES FOR NEXT TIME INCREMENT
221 C
222 1500      DO 1700 I=1,NC
223          DO 1700 J=1,NR
224              IF (SF(I,J).GT.1.0E20) P(I,J)=PO(I,J)
225              D=P(I,J)-PO(I,J)
226              PO(I,J)=P(I,J)
227              F=1.0
228              IF (DL(I,J).EQ.0.0) GO TO 1600
229              IF (ISTEP.GT.2) F=D/DL(I,J)
230              IF (F.GT.5) F=5.0
231              IF (F.LT.0) F=0.0
232 1600      DL(I,J)=D
233 1700      P(I,J)=P(I,J)+D*F
234 C
235 C
236 C      REFINE ESTIMATES OF PRESSURES BY IADI METHOD
237 C
238      ITER=0
239 1800      E=0.0
240      ITER=ITER+1
241 C
242 C      COLUMN CALCULATIONS
243 C
244      DO 3000 II=1,NC
245          I=II
246          IF (MOD(ISTEP+ITER,2).EQ.1) I=NC-I+1
247          DO 2800 J=1,NR
248 C
249 C      CALCULATE B AND G ARRAYS
250 C
251      BB=SF(I,J)/DELTA
252      DD=PO(I,J)+SF(I,J)/DELTA+Q(I,J)
253      AA=0.0
254      CC=0.0
255      IF (J-1) 1900,2000,1900
256 1900      AA=-C(I,J-1,1)
257      BB=BB+C(I,J-1,1)
258 2000      IF (J-NR) 2100,2200,2100
259 2100      CC=-C(I,J,1)
260      BB=BB+C(I,J,1)

```

```

261 2200      IF (I-1) 2500, 2400, 2500
262 2300      BB=BB+C(I-1, J, 2)
263          DD=DD+P(I-1, J)*C(I-1, J, 2)
264 2400      IF (I-NC) 2500, 2600, 2500
265 2500      BB=BB+C(I, J, 2)
266          DD=DD+P(I+1, J)*C(I, J, 2)
267 2600      W=BB-AA*B(J-1)
268 2700      B(J)=CC/W
269          G(J)=(DD-AA*G(J-1))/W
270 2800      CONTINUE
271 C
272 C      RE-ESTIMATE PRESSURES
273 C
274          EN=ABS(P(I, NR)-G(NR))
275          E=AMAX1(E, EN)
276          P(I, NR)=G(NR)
277          N=NR-1
278 2900      HA=G(N)-B(N)*P(I, N+1)
279          EN=ABS(HA-P(I, N))
280          E=AMAX1(E, EN)
281          P(I, N)=HA
282          N=N-1
283          IF (N) 3000, 3000, 2900
284 3000      CONTINUE
285 C
286 C      ROW CALCULATIONS
287 C
288          DO 4100 JJ=1, NR
289              J=JJ
290              IF (MOD(ISTEP+ITER, 2).EQ.1) J=NR-J+1
291                  DO 3900 I=1, NC
292                      BB=SF(I, J)/DELTA
293                      DD=DG(I, J)*SF(I, J)/DELTA+Q(I, J)
294                      AA=0.0
295                      CC=0.0
296                      IF (J-1) 3100, 3200, 3100
297 3100          BB=BB+C(I, J-1, 1)
298                      DD=DD+P(I, J-1)*C(I, J-1, 1)
299 3200          IF (J-NR) 3300, 3400, 3300
300          DD=DD+P(I, J+1)*C(I, J, 1)
301          BB=BB+C(I, J, 1)
302 3400          IF (I-1) 3500, 3600, 3500
303          BB=BB+C(I-1, J, 2)
304          AA=-C(I-1, J, 2)
305 3600          IF (I-NC) 3700, 3800, 3700
306          BB=BB+C(I, J, 2)
307          CC=-C(I, J, 2)
308 3800          W=BB-AA*B(I-1)
309          B(I)=CC/W
310          G(I)=(DD-AA*G(I-1))/W
311 3900          CONTINUE
312 C
313 C      RE-ESTIMATE PRESSURES
314 C
315          EN=ABS(P(NC, J)-G(NC))
316          E=AMAX1(E, EN)
317          P(NC, J)=G(NC)
318          N=NC-1
319 4000          HA=G(N)-B(N)*P(N+1, J)
320          EN=ABS(P(N, J)-HA)
321          E=AMAX1(E, EN)
322          P(N, J)=HA
323          N=N-1
324          IF (N) 4100, 4100, 4000
325 4100          CONTINUE
326          WRITE (*, 50000) ISTEP, ITER, E

```



```

327
328      IF (E.GT.ERROR) GO TO 1800
329 C
330 C
331 C
332 C      WRITE CALCULATED PRESSURE ONTO FILE
333      TIME=TIME/3600.
334      PZAPT=P(1,2)
335      IF(MODE.EQ.0) PZAPT=PZAPT-PAVG
336      WRITE (3,*) TIME,PZAPT
337      WRITE(*,60000) TIME,PZAPT
338 C
339 C      CALCULATE VOLUME INFLUX INTO BOREHOLE
340      I1=1
341      DO 4300 N=1,NLAYER
342      IF (NBC(N).NE.1) GO TO 4300
343      DO 4200 I=1,NV(N)
344 4200      QT=QT+DELTA*C(1,I,2)*(P(2,I)-P(1,I))
345      IF (NBC(N+1).EQ.2) QT=QT+DELTA*C(1,NV(N),1)*(P(1,NV(N))+
346      X      1)-P(1,NV(N)))
347 4300      I1=NV(N)
348 C
349 C      PRINT RESULTS
350 C
351      IF (IOUT*(ISTEP/IOUT).NE.1STEP) GO TO 4500
352 C
353 C      WRITE CURRENT RESULTS ON OUTPUT FILES
354      KOUNT=(NC-1)/6+1
355      DO 4400 K=1,KOUNT
356      I1=1+(K-1)*6
357      I2=K*6
358      IF (I2.GT.NC) I2=NC
359      WRITE (2,62000) TIME,PAVG,QT
360      WRITE (2,64000) (I,I=I1,I2)
361      DO 4400 J=1,NR
362      WRITE (2,66000) J,(P(I,J),I=I1,I2)
363 4400      CONTINUE
364 4500      CONTINUE
365 C
366 C
367 C      WRITE P(Z) AND DP/DR (Z) ONTO FILES
368      DO 4600 J=1,NR
369      PZAPIN=P(1,J)
370      PZAPOUT=P(NC,J)
371      IF(MODE.EQ.0) PZAPIN=PZAPIN-PAVG
372      IF(MODE.EQ.0) PZAPOUT=PZAPOUT-PAVG
373 C
374      WRITE (4,*) ZC(J),PZAPIN
375      WRITE (7,*) ZC(J),PZAPOUT
376      DPDR=(P(3,J)-P(2,J))/(RC(3)-RC(2))
377      WRITE (5,*) ZC(J),DPDR
378 4600      CONTINUE
379 C
380      CLOSE (2)
381      CLOSE (3)
382      CLOSE (4)
383      CLOSE (5)
384      CLOSE (6)
385      CLOSE (7)
386      STOP
387 10000 FORMAT(' ENTER INPUT FILE NAME:')
388 12000 FORMAT(/,' READING FINITE DIFFERENCE MESH DATA FILE',/)
389 14000 FORMAT(/,' BEGIN SIMULATION',/)
390      X      ' ITERATED ALTERNATING DIRECTION IMPLICIT (IADI)',/,
391      X      ' FINITE DIFFERENCE METHOD',/)
392 16000 FORMAT(A)

```

```

393 18000 FORMAT(/, ' ENTER OUTPUT FILE NAME:')
394 20000 FORMAT(/, ' ENTER TOTAL NUMBER OF TIME STEPS AND THE ',/,
395      X      ' NUMBER OF TIME STEPS BETWEEN LISTINGS:')
396 22000 FORMAT(/, ' ENTER SIZE OF TIME STEP (HRS) AND THE ',/,
397      X      ' MAXIMUM PRESSURE CHANGE PER NODE (Pa)',/,
398      X      ' TO CONVERGENCE: ')
399 24000 FORMAT(/, ' ABSOLUTE (0) OR GAGE (1) PRESSURE COMPUTATION?')
400 26000 FORMAT(2E15.6, 3I15)
401 28000 FORMAT(5E15.6)
402 30000 FORMAT(2I15, 2E15.6)
403 32000 FORMAT(/, ' CONSTANT (1) OR TIME VARYING (2) SURFACE PRESSURE')
404 34000 FORMAT(/, ' ENTER UPPER BOUNDARY PRESSURE (Pa):')
405 36000 FORMAT('1', //, ' PSEUDO 3D BAROMETRIC GAS FLOW MODEL', //,
406      X      ' THE INPUT PARAMETERS FOR THIS RUN ARE:', //, ' TIME STEP:',
407      XT30, E16.4, ' Sec', /, ' ERROR LIMIT:', T30, E16.4, ' Pa', /,
408      X      ' VISCOSITY:', T30, E16.4, ' Pa-sec', /,
409      X      ' MEAN BAROMETRIC PRESSURE:', T30, E16.4, ' Pa')
410 38000 FORMAT(//, T5, ' LAYER NO.', T20, ' PERMEABILITY (sq.m.)', T45,
411      X      ' POROSITY', T60, ' HOLE COMPLETION' //)
412 40000 FORMAT(T5, I5, T20, E14.3, T45, E9.3, T60, A20)
413 42000 FORMAT(//, ' RADIAL COORDINATES', //, T5, ' NODE NO.',
414      XT20, ' DISTANCE (m)')
415 44000 FORMAT(T5, I5, T20, F10.2)
416 46000 FORMAT(//, ' VERTICAL COORDINATES', //, T5, ' NODE NO.',
417      XT20, ' LAYER NO.', T35, ' DISTANCE (m)')
418 48000 FORMAT(T5, I5, T20, I5, T35, F10.2)
419 50000 FORMAT(/, ' PRESSURE VS TIME OUTPUT FILE NAME?')
420 52000 FORMAT(/, ' PRESSURE VS DEPTH (WELLFACE) OUTPUT FILE NAME?')
421 54000 FORMAT(/, ' DP/DR VS DEPTH (WELLFACE) OUTPUT FILE NAME?')
422 55000 FORMAT(/, ' PRESSURE VS DEPTH (OUTER BOUNDARY) OUTPUT FILENAME?')
423 56000 FORMAT(2E15.4)
424 58000 FORMAT(' ISTEP=', I4, ' ITER=', I4, ' MAX DELTA P=',
425      XE12.4)
426 60000 FORMAT(/, ' TIME (Hrs) =', E12.4, /,
427      X      ' CAVITY PRESSURE (Pa) =', E12.4, //)
428 62000 FORMAT('1 NODAL PRESSURES (Pa)', //, T5, ' TIME= ',
429      XF10.0, ' secs', /, T5, ' AVG= ', E15.4, ' Pascals', /, T5,
430      X      ' TOTAL VOLUME FLUX INTO WELL= ', E15.4, ' cu.m.')
431 64000 FORMAT(' J=I= ', 6(12.10X))
432 66000 FORMAT(/, 1X, I3, 6E12.5)
433      END

```

C)

```

1 C      +-----+
2 C      +////////// BFD2 (Barometric Finite Difference program) //////////+
3 C      +-----+
4 C
5 C
6 C
7 C      ***** BFD2 IS A DOUBLE PRECISION VERSION OF BFD
8 C
9 C      A COMPUTER PROGRAM WRITTEN FOR:
10 C     EARTH SCIENCE DEPARTMENT
11 C     UNIVERSITY OF CALIFORNIA
12 C     LAWRENCE LIVERMORE NATIONAL LABORATORY
13 C     LIVERMORE, CALIFORNIA
14 C
15 C     WRITTEN BY:
16 C     APPLIED GEOPHYSICS GROUP
17 C     TERRA TEK RESEARCH
18 C     420 WAKARA WAY
19 C     SALT LAKE CITY, UTAH 84108
20 C     OCTOBER, 1984
21 C
22 C     APPLICATION:
23 C     EVALUATION OF THE BAROMETRIC PRESSURE RESPONSE INSIDE A VERTICAL
24 C     CYLINDRICAL HOLE PENETRATING A LAYERED HALF-SPACE. THE HOLE IS
25 C     CASED FROM THE SURFACE TO SOME PRESCRIBED DEPTH. FROM THIS
26 C     DEPTH, THE HOLE CAN BE (1) OPEN, (2) CASED, OR (3) NON-EXISTANT.
27 C     THE HOLE IS SEALED FROM THE ATMOSPHERE AT THE SURFACE. EACH
28 C     LAYER IS GIVEN A PRESCRIBED PERMEABILITY AND POROSITY.
29 C     HOMOGENIETY AND ISOTROPY ARE ASSUMED IN THE LATERAL PLANE.
30 C     2-D FLOW (RADIAL AND VERTICAL) ASSUMED. A NO-FLOW BOUNDARY IS
31 C     ASSUMED AT THE BOTTOM AND OUTER BOUNDARY OF THE RES.
32 C
33 C     CURRENT CONFIGURATION:
34 C     AS PRESENTLY CONFIGURED, BFD CAN DEAL WITH UP TO 10 LAYERS, EACH
35 C     OF WHICH CAN HAVE DIFFERENT THICKNESSES. NUMBER OF VERTICAL NODE
36 C     AND FLOW PROPERTIES. A MAXIMUM OF 200 NODES VERTICALLY AND 20
37 C     NODES RADIALY ARE IMPOSED. RADIAL NODES ARE SPACED IN A
38 C     GEOMETRICALLY INCREASING DISTANCE FROM THE WELLBORE. EG. THE
39 C     FIRST NODE (FROM THE HOLE FACE) IS TWICE THE HOLE RADIUS. THE
40 C     SECOND NODE IS FOUR TIMES THE HOLE RADIUS, AND SO ON. NUMBER OF
41 C     RADIAL NODES IS GOVERNED BY THE USER-SUPPLIED HOLE RADIUS AND
42 C     OUTER "NO-FLOW" BOUNDARY.
43 C
44 C     HARDWARE REQUIREMENTS
45 C     IBM-PC OR IBM-XT W/
46 C     (A) 8087 COPROCESSOR
47 C     (B) 640K MEMORY
48 C     (C) IF NOT AN XT, A HARD DISK EXPANSION UNIT IS RECOMMENDED
49 C     (D) DOT MATRIX PRINTER (FOR FILE OUTPUT)
50 C
51 C     BASED ON:
52 C     PRICKETT, T.A. AND C.G. LONNQUIST
53 C     SELECTED DIGITAL COMPUTER TECHNIQUES FOR GROUNDWATER
54 C     RESOURCE EVALUATION
55 C     ILLINOIS STATE WATER SURVEY, URBANA. BULLETIN 53. 1971
56 C
57 C     DEFINITION OF VARIABLES
58 C
59 C     NSTEPS - NUMBER OF TIME STEPS
60 C     DELTA - SIZE OF TIME STEP (Secs)
61 C     ERRDR - MAXIMUM ALLOWED CONVERGENCE ERROR (Pa)
62 C     NC - NUMBER OF NODAL COLUMNS

```

```

63 C      NR      - NUMBER OF NODAL ROWS
64 C      P(I,J)  - NODAL PRESSURE AT CURRENT TIME STEP (Pa)
65 C      PO(I,J) - NODAL PRESSURE AT PREVIOUS TIME STEP (Pa)
66 C      DL(I,J) - CHANGE IN PRESSURE FROM PREVIOUS TIME STEP (Pa)
67 C      SF(I,J) - STORAGE COEFFICIENT (POROSITY * NODAL AREA / MEAN
68 C              ABSOLUTE PRESSURE) (sq.m./Pa)
69 C      Q(I,J)  - TRUE VOLUMETRIC FLUX PER UNIT THICKNESS (cu.m./sec)
70 C      QT      - TOTAL VOLUME INFLUX INTO WELL (cu.m)
71 C      C(I,J,1) - NODAL CONDUCTIVITY IN Y-DIRECTION (sq. meter/Pa-sec)
72 C      C(I,J,2) - NODAL CONDUCTIVITY IN X-DIRECTION
73 C      PAVG    - AVERAGE ATMOSPHERIC PRESSURE (Pa)
74 C      VIS     - VISCOSITY OF FLUID (Pa-sec)
75 C      PERM(I) - PERMEABILITY OF LAYER I (sq.m.)
76 C      POR(I)  - POROSITY OF LAYER I
77 C
78 C $DEBUG
79 C $STORAGE=2
80 C $NOFLOATCALLS
81 C $LARGE P,PO,SF,C,DL,Q
82 C      REAL*8 P(20,200),PO(20,200),SF(20,200),
83 C      X C(20,200,2),DL(20,200),RC(20),ZC(200),Q(20,200),
84 C      X PERM(10),POR(10)
85 C      REAL*8 B(200),G(200),W,AA,BB,CC,DD,D,F,DELTA,ERROR,E,EN,
86 C      X VIS,TIME,QT,HA
87 C      DIMENSION TEMP(20),NV(10),NBC(10),ZCS(200),PERMS(10),
88 C      X PORS(10),RCS(20)
89 C      CHARACTER*20 INPUT,OUTPUT,PINPUT,FNAME,GNAME,HNAME,COMP(10)
90 C      CHARACTER*20 RNAME
91 C
92 C      SET MS-FORTRAN 8087 MATH ERROR EXCEPTIONS
93 C      (MASK OVERFLOW AND UNDERFLOW)
94 C      CALL LCHRRQ (4927)
95 C
96 C
97 C      READ INPUT PARAMETERS
98 C
99 C      WRITE (*,10000)
100 C      READ (*,16000) INPUT
101 C      OPEN (1,FILE=INPUT,STATUS='OLD')
102 C      WRITE (*,18000)
103 C      READ (*,16000) OUTPUT
104 C      OPEN (2,FILE=OUTPUT,STATUS='NEW')
105 C      WRITE (*,20000)
106 C      READ (*,*) NSTEPS,IOUT
107 C      WRITE (*,22000)
108 C      READ (*,*) DEL,ERR
109 C      CHANGE DELTA TO SECONDS
110 C      DELTA=DBLE(DEL*3600./0)
111 C      ERROR=DBLE(ERR)
112 C      WRITE (*,24000)
113 C      READ (*,*) MDDE
114 C
115 C
116 C      SET UP PRESSURE VS TIME OUTPUT FILE
117 C      WRITE (*,50000)
118 C      READ (*,16000) FNAME
119 C      OPEN (3,FILE=FNAME,STATUS='NEW')
120 C
121 C      SET UP OUTPUT DATA FILES OF PRESSURE VS DEPTH AT WELLBORE,
122 C      PRESSURE VS DEPTH AT OUTER BOUNDARY, AND DP/DR AT WELLBORE.
123 C      AT FINAL TIME ITERATION
124 C      WRITE (*,52000)
125 C      READ (*,16000) GNAME
126 C      WRITE (*,54000)
127 C      READ (*,16000) HNAME
128 C      WRITE (*,56000)

```

```

129      READ (*,16000) RNAME
130 C
131      OPEN (4,FILE=GNNAME, STATUS='NEW')
132      OPEN (5,FILE=HNAME, STATUS='NEW')
133      OPEN (7,FILE=RNAME, STATUS='NEW')
134 C
135 C
136 C      READ INFORMATION FROM DISK FILE (GENERATED BY RUNSFD)
137 C
138 C
139      READ (1,26000) VISS,PAVG,NLAYER,NC,NR
140      READ (1,28000) (ACS(I),I=1,NC)
141      READ (1,28000) (ZCS(I),I=1,NR)
142      READ (1,30000) (NV(I),NBC(I),PERMS(I),PORS(I),I=1,NLAYER)
143 C
144      DO 50 I=1,NC
145          AC(I)=DBLE(ACS(I))
146 50 CONTINUE
147      DO 60 I=1,NR
148          ZC(I)=DBLE(ZCS(I))
149 60 CONTINUE
150      DO 70 I=1,NLAYER
151          PERM(I)=DBLE(PERMS(I))
152          POR(I)=DBLE(PORS(I))
153 70 CONTINUE
154 C
155      VIS=DBLE(VISS)
156 C      SET CONSTANT PRESSURE BOUNDARY (SURFACE)
157 C
158 100 WRITE (*,32000)
159      READ (*,*,ERR=100) NTC
160      IF (NTC.EQ.2) GO TO 300
161 200 WRITE (*,34000)
162      READ (*,*,ERR=200) RUPPER
163      GO TO 400
164 C
165 300 WRITE (*,10000)
166      READ (*,16000,ERR=300) PINPUT
167      OPEN (6,FILE=PINPUT,STATUS='OLD')
168 C
169 400      DO 500 J=1,NLAYER
170          IF (NBC(J).EQ.0) COMP(J)='CASED'
171          IF (NBC(J).EQ.1) COMP(J)='OPEN HOLE'
172          IF (NBC(J).EQ.2) COMP(J)='NO HOLE'
173 500 CONTINUE
174 C
175 C      READ NODE CARDS
176 C
177      WRITE (*,12000)
178      DO 600 J=1,NR
179          DO 600 I=1,NC
180              READ (1,28000) A1,A2,A3,A4,A5
181              C(I,J,1)=DBLE(A1)
182              C(I,J,2)=DBLE(A2)
183              SF(I,J)=DBLE(A3)
184              P(I,J)=DBLE(A4)
185              Q(I,J)=DBLE(A5)
186              IF (MODE.EQ.1) P(I,J)=P(I,J)-DBLE(PAVG)
187              PD(I,J)=P(I,J)
188 600 CONTINUE
189 C
190      REWIND(1)
191      CLOSE (1)
192 C
193 C      SET UPPER BOUNDARY
194      IF (NTC.EQ.2) GO TO 300

```

```

195 700 DO 800 I=2,NC
1 196 SF(1,1)=1.0D25
1 197 P(1,1)=DBLE(PUPPER)
1 198 800 PQ(1,1)=DBLE(PUPPER)
199 GO TO 1100
200 900 DO 1000 I=2,NC
1 201 1000 SF(I,1)=1.025
202 READ (6,58000) PUPPER,ET,ME
203 GO TO 700
204 C
205 C PRINT INPUT PARAMETERS
206 C
207 1100 WRITE (2,36000) DEL,ERR,VISS,PAVG
208 WRITE (2,38000)
209 WRITE (2,40000) (I,PERMS(I),PQRS(I),COMP(I),I=1,NLAYER)
210 WRITE (2,42000)
211 WRITE (2,44000) (I,RCS(I),I=1,NC)
212 WRITE (2,46000)
213 L=1
214 NO 1200 J=1,NR
1 215 IF (J.GT.NV(L)) L=L+1
1 216 WRITE (2,48000) J,L,ZCS(J)
1 217 1200 CONTINUE
218 C
219 C
220 C INITIALIZE OUTPUT DATA FILES
221 WRITE (3,*) NSTEPS
222 WRITE (4,*) NR
223 WRITE (5,*) NR
224 WRITE (7,*) NR
225 C
226 C
227 C START OF SIMULATION
228 C
229 WRITE (*,14000)
230 1300 TIME=0.0D+00
231 QT=0.0D+00
232 DO 4500 ISTEP=1,NSTEPS
1 233 TIME=TIME+DELTA
1 234 C
1 235 C READ UPPER BOUNDARY PRESSURES IF TIME VARYING
1 236 C
1 237 IF (NTC.EQ.1) GO TO 1500
1 238 IF (TIME.LT.DBLE(ETIME)) GO TO 1500
1 239 READ (6,58000) PUPPER,ETIME
1 240 DO 1400 I=2,NC
2 241 1400 P(I,1)=DBLE(PUPPER)
1 242 C
1 243 C PREDICT PRESSURES FOR NEXT TIME INCREMENT
1 244 C
1 245 1500 DO 1700 I=1,NC
2 246 DO 1700 J=1,NR
3 247 IF (SF(I,J).GT.1.0D20) P(I,J)=PQ(I,J)
3 248 D=P(I,J)-PQ(I,J)
3 249 PQ(I,J)=P(I,J)
3 250 F=1.0D+00
3 251 IF (DL(I,J).EQ.0.0D+00) GO TO 1600
3 252 IF (ISTEP.GT.2) F=D/DL(I,J)
3 253 IF (F.GT.5) F=5.0D+00
3 254 IF (F.LT.0) F=0.0D+00
3 255 1600 DL(I,J)=D
3 256 1700 P(I,J)=P(I,J)+D*F
1 257 C
1 258 C
1 259 C REFINED ESTIMATES OF PRESSURES BY IADI METHOD
1 260 C

```

```

1 261          ITER=0
1 262 1800      E=0.0D+00
1 263          ITER=ITER+1
1 264 C
1 265 C      COLUMN CALCULATIONS
1 266 C
1 267          DO 3000 I=1,NC
2 268          I=I
2 269          IF (MOD(ISTEP+ITER,2).EQ.1) I=NC-I+1
2 270          DO 2800 J=1,NA
3 271 C
3 272 C      CALCULATE B AND G ARRAYS
3 273 C
3 274          BB=SF(I,J)/DELTA
3 275          DD=PD(I,J)*SF(I,J)/DELTA+Q(I,J)
3 276          AA=0.0D+00
3 277          CC=0.0D+00
3 278          IF (J-1) 1900,2000,1900
3 279 1900      AA=-C(I,J-1,1)
3 280          BB=BB+C(I,J-1,1)
3 281 2000      IF (J-NA) 2100,2200,2100
3 282 2100      CC=-C(I,J,1)
3 283          BB=BB+C(I,J,1)
3 284 2200      IF (I-1) 2300,2400,2300
3 285 2300      BB=BB+C(I-1,J,2)
3 286          DD=DD+P(I-1,J)*C(I-1,J,2)
3 287 2400      IF (I-NC) 2500,2600,2500
3 288 2500      BB=BB+C(I,J,2)
3 289          DD=DD+P(I+1,J)*C(I,J,2)
3 290 2600      W=BB-AA*B(J-1)
3 291 2700      B(J)=CC/W
3 292          G(J)=(DD-AA*G(J-1))/W
3 293 2800      CONTINUE
2 294 C
2 295 C      RE-ESTIMATE PRESSURES
2 296 C
2 297          EN=DABS(P(I,NA)-G(NA))
2 298          E=DMAX1(E,EN)
2 299          P(I,NA)=G(NA)
2 300          N=NA-1
2 301 2900      HA=G(N)-P(N)*P(I,N+1)
2 302          EN=DABS(HA-P(I,N))
2 303          E=DMAX1(E,EN)
2 304          P(I,N)=HA
2 305          N=N-1
2 306          IF (N) 3000,3000,2900
2 307 3000      CONTINUE
1 308 C
1 309 C      ROW CALCULATIONS
1 310 C
1 311          DO 4100 JJ=1,NA
2 312          J=JJ
2 313          IF (MOD(ISTEP+ITER,2).EQ.1) J=NA-J+1
2 314          DO 3900 I=1,NC
3 315          BB=SF(I,J)/DELTA
3 316          DD=PD(I,J)*SF(I,J)/DELTA+Q(I,J)
3 317          AA=0.0D+00
3 318          CC=0.0D+00
3 319          IF (J-1) 3100,3200,3100
3 320 3100      BB=BB+C(I,J-1,1)
3 321          DD=DD+P(I,J-1)*C(I,J-1,1)
3 322 3200      IF (J-NA) 3300,3400,3300
3 323 3300      DD=DD+P(I,J+1)*C(I,J,1)
3 324          BB=BB+C(I,J,1)
3 325 3400      IF (I-1) 3500,3600,3500
3 326 3500      BB=BB+C(I-1,J,2)

```

```

3 327 AA=-C(I-1,J,2)
3 328 3600 IF (J-NC) 3700,3800,3700
3 329 3700 BB=BB+C(I,J,2)
3 330 CC=-C(I,J,2)
3 331 3800 W=BB-AA*B(I-1)
3 332 B(I)=CC/W
3 333 G(I)=(DD-AA*G(I-1))/W
3 334 3900 CONTINUE
2 335 C
2 336 C RE-ESTIMATE PRESSURES
2 337 C
2 338 EN=DABS(P(NC,J)-G(NC))
2 339 E=DMAX1(E,EN)
2 340 P(NC,J)=G(NC)
2 341 N=NC-1
2 342 4000 HA=G(N)-B(N)*P(N+1,J)
2 343 EN=DABS(P(N,J)-HA)
2 344 E=DMAX1(E,EN)
2 345 ES=SNGL(E)
2 346 P(N,J)=HA
2 347 N=N-1
2 348 IF (N) 4100,4100,4000
2 349 4100 CONTINUE
1 350 WRITE (*,60000) ISTEP,ITER,ES
! 351 C
1 352 IF (E.GT.ERROR) GO TO 1800
1 353 C
1 354 C
1 355 C
1 356 C WRITE CALCULATED PRESSURE ONTO FILE
1 357 TIME=TIME/3600.
1 358 PZAPT=SNGL(P(1,2))
1 359 IF (MODE.EQ.0) PZAPT=PZAPT-PAVG
1 360 WRITE (3,*) TIME,PZAPT
1 361 WRITE (*,62000) TIME,PZAPT
1 362 C
1 363 C CALCULATE VOLUME INFUX INTO BOREHOLE
1 364 I1=1
1 365 DO 4300 N=1,NLAYER
2 366 IF (NBC(N),NE.1) GO TO 4300
2 367 DO 4200 I=I1,NV(N)
3 368 4200 QT=QT+DELTA*C(1,I,2)*(P(2,I)-P(1,I))
2 369 IF (NBC(N+1).EQ.2) QT=QT+DELTA*C(1,NV(N),1)*(P(1,NV(N)-
2 370 X 1)-P(1,NV(N)))
2 371 4300 I1=NV(N)
1 372 C
1 373 C PRINT RESULTS
1 374 C
1 375 IF (IOUT*(ISTEP/IOUT).NE.ISTEP) GO TO 4500
1 376 C
1 377 C WRITE CURRENT RESULTS ON OUTPUT FILES
1 378 KOUNT=(NC-1)/6+1
1 379 DO 4400 K=1,KOUNT
2 380 I1=1+(K-1)*6
2 381 I2=K*6
2 382 IF (I2.GT.NC) I2=NC
2 383 A1=SNGL(TIME)
2 384 A2=SNGL(QT)
2 385 WRITE (2,64000) A1,PAVG,A2
2 386 WRITE (2,66000) (I,I=I1,I2)
2 387 DO 4400 J=1,NR
3 388 DO 4350 I=I1,I2
4 389 TEMP(I)=SNGL(P(I,J))
4 390 4350 CONTINUE
3 391 WRITE (2,68000) J,(TEMP(I),I=I1,I2)
3 392 4400 CONTINUE

```



```

1 393 4500 CONTINUE
394 C
395 C
396 C WRITE P(Z) AND DP/DR (Z) ONTO FILES
397 DO 4600 J=1,NR
1 398 PZAPIN=SNGL(P(1,J))
1 399 PZAPOUT=SNGL(P(NC,J))
1 400 IF (MODE.EQ.0) PZAPIN=PZAPIN-PAVG
1 401 IF (MODE.EQ.0) PZAPOUT=PZAPOUT-PAVG
1 402 C
1 403 WRITE (4,*) ZCS(J),PZAPIN
1 404 WRITE (7,*) ZCS(J),PZAPOUT
1 405 DPDR=SNGL((P(3,J)-P(2,J))/(RC(3)-RC(2)))
1 406 WRITE (5,*) ZCS(J),DPDR
1 407 4600 CONTINUE
408 C
409 CLOSE (2)
410 CLOSE (3)
411 CLOSE (4)
412 CLOSE (5)
413 CLOSE (6)
414 CLOSE (7)
415 STOP
416 10000 FORMAT(' ENTER INPUT FILE NAME:')
417 12000 FORMAT(//,' READING FINITE DIFFERENCE MESH DATA FILE',/)
418 14000 FORMAT(//,' BEGIN SIMULATION',/,
419 X ' ITERATED ALTERNATING DIRECTION IMPLICIT (IADI)',/,
420 X ' FINITE DIFFERENCE METHOD',/)
421 16000 FORMAT(A)
422 18000 FORMAT(//,' ENTER OUTPUT FILE NAME:')
423 20000 FORMAT(//,' ENTER TOTAL NUMBER OF TIME STEPS AND THE ',/,
424 X ' NUMBER OF TIME STEPS BETWEEN LISTINGS:')
425 22000 FORMAT(//,' ENTER SIZE OF TIME STEP (HRS) AND THE ',/,
426 X ' MAXIMUM PRESSURE CHANGE PER NODE (Pa)',/,
427 X ' TO CONVERGENCE: ')
428 24000 FORMAT(//,' ABSOLUTE (0) OR GAGE (1) PRESSURE COMPUTATION?')
429 26000 FORMAT(2E15.6,3I15)
430 28000 FORMAT(5E15.6)
431 30000 FORMAT(2I15,2E15.6)
432 32000 FORMAT(//,' CONSTANT (1) OR TIME VARYING (2) SURFACE PRESSURE')
433 34000 FORMAT(//,' ENTER UPPER BOUNDARY PRESSURE (Pa):')
434 36000 FORMAT('1',//,' PSEUDO 3D BAROMETRIC GAS FLOW MODEL',//,
435 X' THE INPUT PARAMETERS FOR THIS RUN ARE:',//,' TIME STEP:',
436 XT30,E16.4,' Sec',//,' ERROR LIMIT:',T30,E16.4,' Pa',/,
437 X' VISCOSITY:',T30,E16.4,' Pa-sec',/,
438 X' MEAN BAROMETRIC PRESSURE:',T30,E16.4,' Pa')
439 38000 FORMAT(//,T5,' LAYER NO.',T20,' PERMEABILITY (sq.m.)',T45,
440 X' POROSITY',T60,' HOLE COMPLETION')
441 40000 FORMAT(T5,IS,T20,E14.3,T45,E9.3,T60,A20)
442 42000 FORMAT(//,' RADIAL COORDINATES',//,T5,' NODE NO.',
443 XT20,' DISTANCE (m)')
444 44000 FORMAT(T5,IS,T20,F10.2)
445 46000 FORMAT(//,' VERTICAL COORDINATES',//,T5,' NODE NO.',
446 XT20,' LAYER NO.',T35,' DISTANCE (m)')
447 48000 FORMAT(T5,IS,T20,IS,T35,F10.2)
448 50000 FORMAT(//,' PRESSURE VS TIME OUTPUT FILE NAME?')
449 52000 FORMAT(//,' PRESSURE VS DEPTH (WELLFACE) OUTPUT FILE NAME?')
450 54000 FORMAT(//,' DP/DR VS DEPTH (WELLFACE) OUTPUT FILE NAME?')
451 56000 FORMAT(//,' PRESSURE VS DEPTH (OUTER BOUNDARY) OUTPUT FILENAME?')
452 58000 FORMAT(2E15.4)
453 60000 FORMAT(' ISTEP=',I4,' ITER=',I4,' MAX DELTA P=',
454 XE12.4)
455 62000 FORMAT(//,' TIME (Hrs) =',E12.4,/,
456 X ' CAVITY PRESSURE (Pa) =',E12.4,/)
457 64000 FORMAT('1 NODAL PRESSURES (Pa)',//,T5,' TIME= ',
458 XF10.0,' secs',/,T5,' PAVG= ',E15.4,' Pascals',/,T5,

```

```
459      X'TOTAL VOLUME FLUX INTO WELL= ',E15.4,' cu.m.')
```

460 66000 FORMAT(' J=\[= ',6(I2,10X))

461 66000 FORMAT(/,1X,13,6E12.5)

462 END

COMPUTER PROGRAM: KOBRA

KOBRA solves both the forward and inverse problems of a partially cased hole intersecting a layered half-space. The model for both cases is the same as in BFD and BFD2 above. The inverse problem can be carried out with any of the following methods: (1) Monte Carlo, (2) grid search, and (3) ridge regression (Marquardt nonlinear inverse).

C)

```

1 C      ++++++
2 C      ++++++
3 C      ++++++ KOBRA: MODULE 1 ++++++
4 C      ++++++
5 C      ++++++
6 C
7 C      A COMPUTER PROGRAM WRITTEN FOR:
8 C      EARTH SCIENCES DEPARTMENT
9 C      UNIVERSITY OF CALIFORNIA
10 C     LAWRENCE LIVERMORE NATIONAL LABORATORY
11 C     LIVERMORE, CALIFORNIA
12 C
13 C     WRITTEN BY:
14 C     APPLIED GEOPHYSICS GROUP
15 C     TERRA TEK RESEARCH
16 C     420 WAKARA WAY
17 C     SALT LAKE CITY, UTAH 84108
18 C     NOVEMBER, 1984
19 C
20 C     APPLICATION:
21 C     INVERSE1 SOLVES BOTH THE FORWARD AND INVERSE PROBLEM OF THE
22 C     PRESSURE RESPONSE OF A VERTICAL CYLINDRICAL CAVITY EMBEDDED
23 C     IN A LAYERED HALF-SPACE. GIVEN UP TO 10 LAYERS IN THE HALF-
24 C     SPACE, EACH LAYER HAVING A GIVEN COMPLETION CHARACTERISTIC AT
25 C     CAVITY (IE. WELLBORE), THE CAVITY PRESSURE RESPONSE IS COMPUTED
26 C     GIVEN A SURFACE PRESSURE (BAROMETRIC) VARIATION. THE COMPLETION
27 C     CHARACTERISTICS INCLUDE (1) CASED, (2) OPEN-HOLE, AND (3) NO-
28 C     HOLE. WITHIN A GIVEN LAYER (DEFINED AS HAVEN A SINGLE PERMEABILITY
29 C     AND POROSITY), SEVERAL WELL FACE COMPLETIONS ARE ALLOWED. EACH
30 C     OF THESE IS DEFINED AS A SUBDIVISION.
31 C
32 C     MONTE CARLO, GRID SEARCH, OR RIDGE REGRESSION CAN BE USED FOR
33 C     THE INVERSE PROBLEM. CALCULATIONS ARE CARRIED OUT IN THE TIME
34 C     DOMAIN USING AN ITERATED ALTERNATING DIRECTION IMPLICIT FINITE
35 C     DIFFERENCE APPROACH.
36 C     *NOFLOATCALLS
37 C     *DEBUG
38 C     *LARGE
39 C     PROGRAM INVERSE1
40 C
41 C     DIMENSION T(1000),PSURF(1000),PCAVR(1000),PCAVT(1000)
42 C     DIMENSION A(20),REX(20),SIGMAR(20),DELTA(20),JFR(20)
43 C     DIMENSION YFIT(500),SIGMAY(500),B(20),DB(20)
44 C     DIMENSION CHI(20,20),VAR(20,20),COR(20,20)
45 C     DIMENSION NV(10),NBCSUB(200),NVSUB(10,10),NSUB(10)
46 C     DIMENSION KBCSUB(10,10)
47 C
48 C     DOUBLE PRECISION POR(10),XK(10),H(10),HSUB(10,10)
49 C     DOUBLE PRECISION VISC,PAVG,SUM,RC(20),ZC(200)
50 C     CHARACTER*20 ITYPE,NTYPE,GTTYPE,NSEA,TYPE(20)
51 C     CHARACTER*20 HNAME,RNAME,FNAME,COMP,PFLAG,QFLAG
52 C
53 C     COMMON /ZAP/ JFR,NTERMS,NFREE,NPTS
54 C     COMMON /DAT/PSURF,PCAVT,DT,ER,PAVG,PFLAG
55 C     COMMON/STUFF/MSTART
56 C     COMMON/STUFF2/QFLAG
57 C     COMMON/GEOM/NLAYER,NSUB,NV,NVSUB,NBCSUB,RC,ZC,NR,NC,RMIN
58 C     COMMON/DER/LPTS
59 C
60 C     SET MS-FORTRAN 8087 MATH ERROR EXCEPTIONS
61 C     (MASK OVERFLOW AND UNDERFLOW)
62 C     CALL LCWRQQ(4927)

```

```

63 C
64      PI=3.141592654
65      TWOPI=2.0*PI
66 C
67 C
68 C      INITIALIZE PARAMETER INDICES
69      DO 100 J=1,20
1 70          JFR(J)=J
1 71      100      CONTINUE
72 C
73      PI=3.14159265
74      ID1=0
75      ID2=0
76 C
77 C      SET VALUES FOR GAS VISCOSITY (pa.sec) AND MEAN GAS PRESSURE (pa)
78      VISC=2.0D-05
79      PAVG=8.5D04
80 C
81 C      USE DEFAULT INITIAL FORMATION PRESSURE (85 kPa) OR
82 C      USE TIME-AVERAGED:
83 C          (1) MEASURED SURFACE PRESSURE FOR FORWARD PROBLEM
84 C          (2) MEASURED DOWNHOLE PRESSURE FOR INVERSE PROBLEM
85      WRITE (ID2,20000)
86      READ (ID1,70000) STYPE
87 C
88 C      ENTER HOLE-FORMATION GEOMETRY, AND FORMATION FLOW PARAMETERS
89 C
90 C      INPUT WELL RADIUS AND OUTER RADIUS
91      WRITE (ID2,10000)
92      READ (ID1,*) RMIN
93      RMAX=1000.0
94 C
95 C
96 C      INPUT HALF-SPACE PARAMETERS
97      WRITE (ID2,23000)
98      READ (ID1,*) NLAYER
99      NTERMS=2*NLAYER
100 C
101      KOUNT=1
102      SUM=0.0D0
103      DO 400 J=1,NLAYER
1 104 C
1 105 C      SPECIFY PERMEABILITY, POROSITY OF CURRENT LAYER
1 106          WRITE (ID2,26000) J
1 107          READ (ID1,*) A2,A3
1 108          XK(J)=DBLE(A2)
1 109          POR(J)=DBLE(A3)
1 110 C
1 111 C      SPECIFY SUBDIVISIONS AT WELLFACE WITHIN CURRENT LAYER
1 112          WRITE (ID2,13000) J
1 113          READ (ID1,*) NSUB(J)
1 114 C
1 115          LNS=0
1 116          H(?)=0.0D0
1 117          DO 300 K=1,NSUB(J)
2 118              WRITE (ID2,14000) K,J
2 119              READ (ID1,*) HS
2 120              HSUB(J,K)=DBLE(HS)
2 121              H(J)=H(J)+HSUB(J,K)
2 122              WRITE (ID2,12000) K,J
2 123              READ (ID1,*) MBC
2 124              KBCSUB(J,K)=MBC
2 125              WRITE (ID2,11000) K,J
2 126              READ (ID1,*) NVSUB(J,K)
2 127              DO 200 L=1,NVSUB(J,K)
3 128                  NBCSUB(KOUNT)=MBC

```

```

3 129      KBCSUB(J,K)=MBC
3 130      KOUNT=KOUNT+1
3 131      200  CONTINUE
2 132      NNS=NNS+NVSUB(J,K)
2 133      300  CONTINUE
1 134      NV(J)=NNS
1 135 C
1 136 C      LOAD PERMEABILITIES AND POROSITIES INTO PARAMETER ARRAY
1 137      A(2*J-1)=SNGL(XK(J))
1 138      A(2*J) =SNGL(POR(J))
1 139      AEX(2*J-1)=A(2*J-1)
1 140      AEX(2*J)=A(2*J)
1 141      400  CONTINUE
142 C
143 C      COMPUTE NODAL COORDINATES
144 C
145      RC(1)=0.0D0
146      I=1
147      500  I=I+1
148      RC(I)=DBLE(RMIN*2.0**(I-1))
149      IF (RC(I)-DBLE(RMAX)) 500,600,600
150      600  RC(I)=DBLE(RMAX)
151      NC=I
152      D=0
153      ZC(1)=0.0D0
154      NR=1
155      DO 900 I=1,NLAYER
1 156 C
1 157      DO 800 J=1,NSUB(I)
2 158      UD=HSUB(I,J)/NVSUB(I,J)
2 159 C
2 160      DO 700 K=1,NVSUB(I,J)
3 161      D=D+DD
3 162      NR=NR+1
3 163      700  ZC(NR)=DBLE(D)
2 164      800  CONTINUE
1 165 C
1 166      900  NV(I)=NR
167      KBCSUB(NR)=MBC
168 C
169 C      PRINT OUT HOLE-FORMATION GEOMETRY
170      WRITE (ID2,15000)
171      WRITE (ID2,16000)
172      DO 1100 J=1,NLAYER
1 173      DO 1000 K=1,NSUB(J)
2 174      IF (KBCSUB(J,K).EQ.0) COMP='CASED'
2 175      IF (KBCSUB(J,K).EQ.1) COMP='OPEN HOLE'
2 176      IF (KBCSUB(J,K).EQ.2) COMP='NO HOLE'
2 177      WRITE (ID2,17000) J,K,HSUB(J,K),NVSUB(J,K),COMP

```

Page

```

2 178 1000  CONTINUE
1 179 1100  CONTINUE
180 C
181 C
182 C      INPUT FINITE DIFFERENCE ITERATION CONVERGENCE CRITERION
183      WRITE (ID2,18000)
184      READ (ID1,*) ER
185 C
186 C      SPECIFY WHETHER OR NOT TO PRINT OUT COMPUTED PRESSURES
187 C      (EITHER IN FORWARD OR INVERSE CALCULATIONS) AT EACH
188 C      TIME STEP. NOTE: THIS INFO. WILL BE STORED ON A DISK FILE
189 C      ANYWAY...IF YOU DON'T WANT A LOT OF HARDCOPY, SPECIFY
190 C      NO!
191      WRITE(ID2,72000)
192      READ(ID1,70000) PFLAG

```

```

193 C
194 C SPECIFY EITHER FORWARD OR INVERSE CALCULATION
195 C FORWARD: COMPUTE THEORETICAL PRESSURE RESPONSE IN THE TIME
196 C DOMAIN GIVEN INPUT SYSTEM PARAMETERS.
197 C INVERSE: COMPUTE SYSTEM PARAMETERS GIVEN THEORETICAL
198 C PRESSURE RESPONSE IN THE TIME DOMAIN.
199 C
200 WRITE (ID2,32000)
201 READ (ID1,70000) NTYPE
202 IF (NTYPE.EQ.'I') GO TO 2100
203 C
204 C
205 C ----- FORWARD SOLUTION -----
206 C
207 C
208 C ENTER MEASURED SURFACE PRESSURE DATA FILENAME
209 WRITE (ID2,44000)
210 READ (ID1,70000) RNAME
211 OPEN (7, FILE=RNAME, STATUS='OLD')
212 C
213 C
214 C ENTER COMPUTED DOWNHOLE PRESSURE DATA FILENAME
215 WRITE (ID2,45000)
216 READ (ID1,70000) FNAME
217 OPEN (8, FILE=FNAME, STATUS='NEW')
218 C
219 C CHECK IF SURFACE DATA IS IN PSI OR PA
220 WRITE (ID2,19000)
221 READ (ID1,70000) ITYPE
222 IF (ITYPE.EQ.'PSI') CONV=6.8947E03
223 IF (ITYPE.EQ.'PA') CONV=1.0
224 C
225 C ADD NOISE TO SYSTEM RESPONSE?
226 XNS=0.0
227 MODE=0
228 WRITE (ID2,53000)
229 READ (ID1,56000) NOIS
230 IF (NOIS.NE.'Y') GO TO 1200
231 MODE=1
232 WRITE (ID2,54000)
233 READ (ID1,*) XNS
234 C
235 C
236 1200 WRITE (ID2,21000)
237 READ (ID1,*) NBUMP
238 C
239 C READ DATA FROM INPUT DATA FILE
240 MBUMP=0
241 READ (7,*) NT
242 NPTS=NT
243 K=0
244 SS=0.0
245 DO 1400 J=1,NT
1 246 READ (7,*) TT,PP
1 247 MBUMP=MBUMP+1
1 248 IF (MBUMP.NE.NBUMP) GO TO 1400
1 249 K=K+1
1 250 MBUMP=0
1 251 T(K)=TT
1 252 PSURF(K)=CONV*PP
1 253 SS=SS+CONV*PP
1 254 1400 CONTINUE
255 C
256 NPTS=K-1
257 NT=NPTS
258 PAVTOP=SS/FLOAT(NPTS)

```

```

259      IF (GTYPE.EQ.'B') PAVG=DBLE(PAVTOP)
260 C
261 C
262      NTERMS=2*NLAYR
263      NVAR=NTERMS
264      NFREE=NPTS-NTERMS
265      NFR=NFREE
266 C
267 C      CONVERT DELTA TIME TO SECONDS
268      DT=3600.*(T(2)-T(1))
269 C
270 C      COMPUTE THEORETICAL DOWNHOLE PRESSURE RESPONSE
271      DO 1800 J=1,NT
272          PCAVT(J)=FUNCTN(T,J,A)
1 273 1800      CONTINUE
274 C
275 C      STORE THEORETICAL PRESSURE RESPONSE ON FILE
276      WRITE (8,*) NT
277      DUMMY=.21289
278      DO 2000 J=1,NT
1 279          CALL RAN(DUMMY)
1 280          ZIP=DUMMY
1 281          PCAVT(J)=PCAVT(J)+XNS*(ZIP-.5)
1 282          SIGMAY(J)=XNS
1 283          WRITE (8,*) T(J),PCAVT(J)
1 284 2000      CONTINUE
285 C
286      CLOSE (8,STATUS='KEEP')
287 C
288      GO TO 5500
289 C
290 C      ----- END OF FORWARD SOLUTION -----
291 C
292 C      ----- INVERSE SOLUTION -----
293 C
294 C      OPEN MEASURED SURFACE PRESSURE DATA FILE
295 2100 WRITE (ID2,44000)
296      READ (ID1,70000) RNAME
297      OPEN (7, FILE=RNAME, STATUS='OLD')
298 C
299 C      OPEN MEASURED DOWNHOLE PRESSURE DATA FILE
300      WRITE (ID2,45000)
301      READ (ID1,70000) HNAME
302      OPEN (8, FILE=HNAME, STATUS='OLD')
303 C
304 C      OPEN COMPUTED DOWNHOLE PRESSURE DATA FILE
305      WRITE (ID2,44500)
306      READ (ID1,70000) FNAME
307      OPEN (9, FILE=FNAME, STATUS='NEW')
308 C
309 C      CHECK IF SURFACE DATA IS IN PSI OR PA
310      WRITE (ID2,19000)
311      READ (ID1,70000) GTYPE
312      IF (GTYPE.EQ.'PSI') CONV=6.8947E03
313      IF (GTYPE.EQ.'PA') CONV=1.0
314 C
315 C      SPECIFY MAXIMUM TIME
316      WRITE (ID2,77000)
317      READ (ID1,*) TMAX
318 C
319 C      READ TIME AND PRESSURE DATA
320 2150 WRITE (ID2,21000)
321      READ (ID1,*) NEUMP
322 C
323      READ (7,*) NT
324      READ (8,*) NT

```



```

325      NPTS=NT
326      K=0
327      MBUMP=0
328      SS1=0.0
329      SS2=0.0
330 C
331      DO 2400 J=1,NT
1 332      READ (7,*) TT1,PP1
1 333      READ (8,*) TT2,PP2
1 334      MBUMP=MBUMP+1
1 335      IF (MBUMP.NE.NBUMP) GO TO 2400
1 336      K=K+1
1 337      MBUMP=0
1 338      T(K)=TT2
1 339      PSURF(K)=CONV*PP1
1 340      PCAVR(K)=CONV*PP2
1 341      SIGMAY(K)=XNS
1 342      SS1=SS1+CONV*PP1
1 343      SS2=SS2+CONV*PP2
1 344      IF (T(K).GT.TMAX) GO TO 2450
1 345 2400      CONTINUE
346 C
347 2450 NPTS=K
348      NT=NPTS
349      WRITE(9,*) NPTS
350      LPTS=NPTS
351      DT=3600.*(T(2)-T(1))
352      PAVTOP=SS1/FLOAT(L)
353      PAVBOT=SS2/FLOAT(K)
354      DELP=PAVBOT-PAVTOP
355      IF (GTYPE.EQ.'B') PAVG=DBLE(PAVBOT)
356 C
357 C      REFERENCE PRESSURE MEASUREMENTS TO COMMON ELEVATION?
358      WRITE(ID2,73000)
359      READ(ID1,70000) ITYPE
360      IF (ITYPE.NE.'Y') GO TO 2650
361 C
362      WRITE(ID2,75000) PAVTOP,PAVBOT
363 C      ESTIMATE ELEVATION DIFFERENCE BETWEEN SURFACE SENSOR AND
364 C      IN-HOLE SENSOR
365      DEPSENS=16000.*DELP/(PAVTOP+PAVBOT)
366      WRITE(ID2,71000) DEPSENS
367 C
368 C      REFERENCE PRESSURE DATA (SURFACE AND DOWNHOLE) TO COMMON DEPT-S
369      DO 2600 J=1,NPTS
1 370      PSURF(J)=PSURF(J)+DELP
1 371 2600      CONTINUE
372 C
373 C      INPUT ESTIMATED UNCERTAINTY IN DATA
374 C      MODE=0...MEASURED DOWNHOLE DATA IS EXACT
375 C      MODE=1...MEASURED DOWNHOLE DATA HAS A "ONE-SIGMA" = UNCEA
376 2650 WRITE (ID2,24000)
377      READ (ID1,*) MODE
378      IF (MODE.EQ.0) GO TO 2800
379      WRITE (ID2,25000)
380      READ (ID1,*) UNCEA
381      DO 2700 J=1,NPTS
1 382      SIGMAY(J)=UNCEA
1 383 2700      CONTINUE
384 C
385 C      ENCODE PARAMETER SET
386 C      INPUT NO. FREE PARAMETERS IN MODEL
387 2800 WRITE (ID2,33000)
388      READ (ID1,*) NFR
389      NFREE=NPTS-NFR
390      NFRP1=NFR+1

```

```

391      NTERMS=2*NLAYER
392 C
393 C      SPECIFY WHICH PARAMETERS IN MODEL ARE FREE PARAMETERS...
394 C      IE. ALLOWED TO VARY TO FIT THE DATA
395      DO 2900 J=1,NFR
1 396      WRITE (ID2,34000) J
1 397      READ (ID1,*) JFR(J)
1 398      TYPE (J)='FREE'
1 399 2900      CONTINUE
400 C
401 C      ASSIGN REMAINDER OF ENCODED MODEL PARAMETERS
402      DO 3000 J=NFRP1,NTERMS
1 403      K=J-NFRP1+1
1 404      WRITE (ID2,35000) K
1 405      READ (ID1,*) JFR(J)
1 406      TYPE (J)='FIXED'
1 407 3000      CONTINUE
408 C
409 C      SPECIFY VALUES OF FIXED MODEL PARAMETERS
410      DO 3100 J=NFRP1,NTERMS
1 411      A(J)=AEX(JFR(J))
1 412 3100      CONTINUE
413 C
414 C      SPECIFY STARTING VALUES OF FREE PARAMETERS
415      DO 3200 J=1,NFR
1 416      A(J)=AEX(JFR(J))
1 417 3200      CONTINUE
418 C
419 C      PRINT OUT MODEL PARAMETER SUMMARY
420      WRITE (ID2,27000)
421      DO 3300 J=1,NTERMS
1 422      WRITE (ID2,28000) JFR(J),AEX(JFR(J)),TYPE(J)
1 423 3300      CONTINUE
424 C
425 C      ENTER BEGINNING TIME FOR WHICH DIFFERENCE BETWEEN MODEL AND
426 C      DATA IS TO BE MINIMIZED.
427      WRITE (ID2,31000)
428      READ (ID1,*) TST
429 C
430 C      DETERMINE STARTING INDEX FOR DATA-MODEL COMPARISON
431      DO 3400 J=1,NPTS
1 432      IF (T(J).GE.TST) GO TO 3500
1 433 3400      CONTINUE
434 3500 MSTART=J
435      JPTS=NPTS-MSTART+1
436 C
437 C      DEFINE NUMBER OF DEGREES OF FREEDOM BASED ON NUMBER OF DATA
438 C      POINTS IN MODEL-DATA COMPARISON AND NUMBER OF FREE PARAMETERS
439 C      IN MODEL.
440      NFREE=JPTS-NFR
441 C
442      WRITE (ID2,50000)
443      READ (ID1,70000) MC
444 C
445      IF (MC.NE.'Y') GO TO 4100
446 C
447 C      ----- BEGIN MONTE CARLO SEARCH -----
448 C
449      KOUNT=0
450      TEMP=1.0E20
451      NPRN=NTERMS/4
452 C      INPUT PARAMETER RANGE FOR MONTE CARLO SEARCH
453      DO 3600 J=1,NFR
1 454      WRITE (ID2,51000) JFR(J)
1 455      READ (ID1,*) B1,B2
1 456      B(J)=B1

```

```

1 457          DB(J)=D2-B1
1 458 3600      CONTINUE
459 C
460 C      INPUT NO. TIMES DICE ARE TO BE THROWN
461          WRITE (ID2,22000)
462          READ (ID1,*) KMAX
463          DUMMY=C.21289
464 C
465 C      THROW THE DICE
466 3700      DO 3800 J=1,NFR
467          CALL RAN(DUMMY)
1 468          A(J)=B(J)+DUMMY*DB(J)
1 469 3800      CONTINUE
470 C
471          DO 3900 J=1,NPTS
1 472          PCAVT(J)=FUNCTN(T,J,A)
1 473 3900      CONTINUE
474 C
475          KOUNT=KOUNT+1
476          IF (KOUNT.GT.KMAX) GO TO 5500
477          CHISQR=FCHISQ(PCAVR,SIGMA,NPTS,NFREE,MODE,PCAVT)
478          IF (CHISQR.GT.TEMP) GO TO 3700
479          PROBAB=PCHISQ(CHISQR,NFREE)
480 C
481          WRITE (ID2,63000) CHISQR
482          WRITE (ID2,64000) PROBAB
483          DO 4000 LL=1,NPRN
1 484          KBDT=(LL-1)*4+1
1 485          KTOP=LL*4
1 486          IF (LL.EQ.NPRN) KTOP=NTERMS
1 487          WRITE (ID2,62000)
1 488          WRITE (ID2,58000) (JFR(K),K=KBDT,KTOP)
1 489          WRITE (ID2,57000) (A(K),K=KBDT,KTOP)
1 490 4000      CONTINUE
491 C
492          TEMP=CHISQR
493          GO TO 3700
494 C      ----- END OF MONTE CARLO SEARCH -----
495 C
496 C      ----- BEGIN NON-LINEAR INVERSION -----
497 C
498 C      SET UP PARAMETERS FOR NON-LINEAR INVERSION
499 4100 NPTS=NT
500          FLAMDA=0.001
501 C
502 C      SPECIFY FRACTION OF START VALUE TO BE INCREMENT
503          WRITE (ID2,48000)
504          READ (ID1,*) XFRAC
505 C
506 C      INITIALIZE PARAMETER INCREMENTS
507          DO 4200 J=1,NFR
1 508          DELTAA(J)=XFRAC*A(J)
1 509 4200      CONTINUE
510 C
511 C      SPECIFY PRECISION OF CHISQR MINIMUM
512          WRITE (ID2,69000)
513          READ (ID1,*) PRES
514 C
515 C      PRINT OUT A LIMITED HISTORY OF THE PARAMETER SEARCH.
516 C      THIS WILL ALLOW THE USER TO IDENTIFY AT ANY GIVEN TIME
517 C      DURING EXECUTION JUST EXACTLY WHERE THE INVERSION IS.
518          WRITE(ID2,76000)
519          READ(ID1,70000) OFLAG
520 C
521 C      SPECIFY EITHER MARQUARDT OR GRID SEARCH
522          WRITE (ID2,47000)

```

```

523      READ (ID1,70000) NSEA
524 C
525 C      WRITE HEADER
526      IF (NSEA.EQ.'M') WRITE (ID2,41000)
527      IF (NSEA.EQ.'G') WRITE (ID2,42000)
528 C
529 C      COMMENCE ITERATION SEQUENCE...STOP WHEN CHISQR IS MINIMUM
530      TEMP=1.0
531      DO 5200 J=1,200
1 532      WRITE (ID2,59000) J
1 533      IF (NSEA.EQ.'G') GO TO 4500
1 534      CALL CURFIT(T,PCAVR,SIGMAV,NPTS,NFR,MODE,A,DELTA,VAR,FLAMD
1 535      Y      PCAVT,CHISQR,NVAR)
1 536      DO 4400 K=1,NFR
2 537      SIGMAA(K)=SQRT(VAR(K,K))
2 538      DO 4300 L=1,K
3 539      CDR(K,L)=SQRT(VAR(K,L)**2/(VAR(K,K)*VAR(L,L)))
3 540      4300      CONTINUE
2 541      4400      CONTINUE
1 542      GO TO 4600
1 543      4500      CALL GRIDLS(T,PCAVR,SIGMAV,NPTS,NFR,MODE,A,DELTA,SIGMAA,
1 544      X      PCAVT,CHISQR)
1 545 C
1 546      4600      PROBAB=PCHISQ(CHISQR,NFREE)
1 547      NPRN=NTERMS/4
1 548      IF (NTERMS.LT.4) NPRN=1
1 549 C
1 550      WRITE (ID2,63000) CHISQR
1 551      WRITE (ID2,64000) PROBAB
1 552      WRITE (ID2,62000)
1 553      DO 4700 LL=1,NPRN
2 554      KBOT=(LL-1)*4+1
2 555      KTOP=LL*4
2 556      IF (KTOP.GT.NTERMS) KTOP=NTERMS
2 557      MDIF=KTOP-KBOT+1
2 558      IF (MDIF.EQ.1) GOTO 4650
2 559      IF (MDIF.EQ.2) GOTO 4660
2 560      IF (MDIF.EQ.3) GOTO 4670
2 561      WRITE (ID2,58000) (JFR(K),K=KBOT,KTOP)
2 562      WRITE (ID2,57000) (A(K),K=KBOT,KTOP)
2 563      WRITE (ID2,61000) (SIGMAA(K),K=KBOT,KTOP)
2 564      GOTO 4700
2 565      4650      WRITE (ID2,58003) (JFR(K),K=KBOT,KTOP)
2 566      WRITE (ID2,57003) (A(K),K=KBOT,KTOP)
2 567      WRITE (ID2,61003) (SIGMAA(K),K=KBOT,KTOP)
2 568      GOTO 4700
2 569      4660      WRITE (ID2,58002) (JFR(K),K=KBOT,KTOP)
2 570      WRITE (ID2,57002) (A(K),K=KBOT,KTOP)
2 571      WRITE (ID2,61002) (SIGMAA(K),K=KBOT,KTOP)
2 572      GOTO 4700
2 573      4670      WRITE (ID2,58001) (JFR(K),K=KBOT,KTOP)
2 574      WRITE (ID2,57001) (A(K),K=KBOT,KTOP)
2 575      WRITE (ID2,61001) (SIGMAA(K),K=KBOT,KTOP)
2 576      4700      CONTINUE
1 577 C
1 578      WRITE (ID2,67000)
1 579      IF (NSEA.EQ.'G') GO TO 5100
1 580      WRITE (ID2,65000)
1 581      DO 4800 K=1,NFR
2 582      DO 4750 L=1,K
3 583      WRITE (ID2,68000) K,L,CDR(K,L)
3 584      4750      CONTINUE
2 585      4800      CONTINUE
1 586      WRITE (ID2,67000)
1 587      DO 5000 K=2,NFR
2 588      DO 4900 L=1,K-1

```

```

3 589 R=CDR(K,L)
3 590 CORREL=1.0- PCORRE(R,JPTS,NFR)
3 591 WRITE (ID2,66000) JFR(K),JFR(L),CORREL
3 592 4900 CONTINUE
2 593 5000 CONTINUE
1 594 C
1 595 5100 WRITE (ID2,67000)
1 596 C
1 597 RAT=ABS((TEMP-CHISQR)/TEMP)
1 598 IF (RAT,LF,PRES) GO TO 5300
1 599 TEMP=CHISQR
1 600 WRITE (ID2,68000)
1 601 5200 CONTINUE
602 C
603 5300 DO 5400 J=1,NPTS
1 604 WRITE (9,*) T(J),PCAVT(J)
1 605 5400 CONTINUE
606 5500 STOP
507 10000 FORMAT(/,' ENTER WELL RADIUS (m)')
608 11000 FORMAT(/,' ENTER NO. VERTICAL NODES FOR SUBDIVISION ',I2,
609 X ' OF LAYER ',I2)
610 12000 FORMAT(/,' ENTER LAYER BOUNDARY CONDITION AT WELL BORE'//,
611 X ' FOR SUBDIVISION ',I2, ' OF LAYER ',I2,/,
612 X ' (0=CASED, 1=OPEN HOLE, 2=NO HOLE)')
613 13000 FORMAT(/,' ENTER NUMBER OF SUBDIVISIONS IN LAYER NO. ',I2)
614 14000 FORMAT(/,' ENTER THICKNESS (m) OF SUBDIVISION ',I2 ' OF '
615 X ' LAYER ',I2)
616 15000 FORMAT(/,' 5X,' LAYERED HALF-SPACE GEOMETRY'//)
617 16000 FORMAT(/,' LAYER SUB-LAYER SUB-LAYER SUB-LAYER COMPLETION',
618 X ' NO. NO. THICKNESS NODES TYPE'//)
619 17000 FORMAT(1H,3X,I2,5X,I2,5X,F8.2,5X,I2,8X,A10)
620 18000 FORMAT(/,' ENTER FINITE DIFFERENCE CONVERGENCE CRITERIA (Pa)')
621 19000 FORMAT(/,' DATA IS PSI OR PA? (ENTER PSI OR PA)')
622 20000 FORMAT(/,' DEFAULT P0 (85KPa) = A, COMPUTED = B')
623 21000 FORMAT(/,' READ EVERY Nth DATA POINT - ENTER N')
624 22000 FORMAT(/,' NO. TIMES DICE ARE TO BE THROWN? (14'')
625 23000 FORMAT(/,' ENTER NUMBER OF LAYERS')
626 24000 FORMAT(/,' DATA MEASUREMENT IS EXACT : 0, OTHERWISE : 1')
627 25000 FORMAT(/,' ENTER DATA UNCERTAINTY (Pa)')
628 26000 FORMAT(/,' ENTER PERMEABILITY (D) AND POROSITY OF '
629 X ' LAYER ',I2)
630 27000 FORMAT(/,' MODEL PARAMETER SUMMARY')
631 28000 FORMAT(/,I2,5X,E12.4,5X,A6)
632 31000 FORMAT(/,' ENTER START TIME (Hrs.) FOR COMPARISON ',
633 X ' BETWEEN MODEL AND DATA')
634 32000 FORMAT(/,' FORWARD=F, INVERSE=I')
635 33000 FORMAT(/,' ENTER NO. FREE PARAMETERS IN MODEL ')
636 34000 FORMAT(/,' ENTER MODEL PARAMETER CORRESPONDING TO FREE '
637 X'PARAMETER ',I2)
638 35000 FORMAT(/,' ENTER MODEL PARAMETER CORRESPONDING TO FIXED '
639 X'PARAMETER ',I2)
640 41000 FORMAT(/,' 1X,' RIDGE REGRESSION ITERATION SEQUENCE'//)
641 42000 FORMAT(/,' 1X,' GRID SEARCH ITERATION SEQUENCE'//)
642 44000 FORMAT(/,' ENTER MEASURED SURFACE PRESSURE DATA FILE NAME')
643 44500 FORMAT(/,' ENTER COMPUTED DOWNHOLE PRESSURE DATA FILE NAME')
644 45000 FORMAT(/,' ENTER MEASURED DOWNHOLE PRESSURE DATA FILE NAME')
645 47000 FORMAT(/,' MARQUARDT=M, GRID=G')
646 48000 FORMAT(/,' ENTER FRACTION OF INITIAL A TO BE DELTA-A')
647 50000 FORMAT(/,' MONTE CARLO SEARCH? (Y/N)')
648 51000 FORMAT(/,' ENTER MIN,MAX VALUES FOR MODEL PARAMETER ',I2)
649 53000 FORMAT(/,' ADD NOISE TO COMPUTED SYSTEM RESPONSE? (Y/N)')
650 54000 FORMAT(/,' ENTER NOISE LEVEL (PRESSURE UNITS)')
651 56000 FORMAT(A1)
652 57000 FORMAT(1H,4(E11.4,10X))
653 57001 FORMAT(1H,3(E11.4,10X))
654 57002 FORMAT(1H,2(E11.4,10X))

```

```

655 57003 FORMAT(1H,1(E11.4,10X))
656 58000 FORMAT(1H,4(' MODEL PARAM.',I2,5X),/)
657 58001 FORMAT(1H,3(' MODEL PARAM.',I2,5X),/)
658 58002 FORMAT(1H,2(' MODEL PARAM.',I2,5X),/)
659 58003 FORMAT(1H,1(' MODEL PARAM.',I2,5X),/)
660 59000 FORMAT(1X) ***** ITERATION ',I3,' *****',/)
661 60000 FORMAT(1X) ' *****',/)
662 61000 FORMAT(1H,4(' ',E11.4,' '),8X),/)
663 61001 FORMAT(1H,3(' ',E11.4,' '),8X),/)
664 61002 FORMAT(1H,2(' ',E11.4,' '),8X),/)
665 61003 FORMAT(1H,1(' ',E11.4,' '),8X),/)
666 62000 FORMAT(1H,15X,' PARAMETERS AND UNCERTAINTIES',/)
667 63000 FORMAT(1H,/, ' REDUCED CHI-SQUARE = ',E12.4)
668 64000 FORMAT(1H,/, ' PROBABILITY OF CORRECT PARENT FUNCTION = ',
669 X E12.4,/)
670 65000 FORMAT(1H,10X,' FREE PARAMETER CORRELATION MATRIX',/)
671 66000 FORMAT(1H, ' PROBABILITY THAT A(' ,I1,') IS CORRELATED WITH '
672 X 'A(' ,I1,') IS ',E12.4,/)
673 67000 FORMAT(1H )
674 68000 FORMAT(1H,14X,I2,2X,I2,2X,F12.4)
675 69000 FORMAT(/, ' ENTER CONVERGENCE CRITERION')
676 70000 FORMAT(A)
677 71000 FORMAT(/, ' ELEVATION DIFFERENCE (m) BETWEEN SURFACE AND IN-HOLE
678 X /, ' SENSORS IS ',F5.2)
679 72000 FORMAT(/, ' PRINT INFO. AT EACH TIME STEP? (Y/N)')
680 73000 FORMAT(/, ' REFERENCE SENSOR DATA TO COMMON ELEVATION? (Y/N)')
681 75000 FORMAT(/, ' AVERAGE MEASURED SURFACE PRESSURE IS ',E16.6,/,
682 X ' AVERAGE MEASURED DOWNHOLE PRESSURE IS ',E16.6)
683 76000 FORMAT(/, ' PRINT OUT LIMITED HISTORY OF INVERSION?(Y/N)')
684 77000 FORMAT(/, ' ENTER MAXIMUM TIME (hrs)')
685 END

686 FUNCTION FUNCTN(T, KK, AC)
687 C
688 C
689 C DEFINITION OF VARIABLES
690 C
691 C NSTEPS - NUMBER OF TIME STEPS
692 C DELTA - SIZE OF TIME STEP (Secs)
693 C ERROR - MAXIMUM ALLOWED CONVERGENCE ERROR (Pa)
694 C NC - NUMBER OF NODAL COLUMNS
695 C NR - NUMBER OF NODAL ROWS
696 C P(I,J) - NODAL PRESSURE AT CURRENT TIME STEP (Pa)
697 C PO(I,J) - NODAL PRESSURE AT PREVIOUS TIME STEP (Pa)
698 C DL(I,J) - CHANGE IN PRESSURE FROM PREVIOUS TIME STEP (Pa)
699 C SF(I,J) - STORAGE COEFFICIENT (POROSITY * NODAL AREA / MEAN
700 C ABSOLUTE PRESSURE) (sq.m./Pa)
701 C Q(I,J) - TRUE VOLUMETRIC FLUX PER UNIT THICKNESS (cu.m./sec)
702 C QT - TOTAL VOLUME INFLUX INTO WELL (cu.m)
703 C C(I,J,1)- NODAL CONDUCTIVITY IN Y-DIRECTION (sq. meter/Pa-sec)
704 C C(I,J,2)- NODAL CONDUCTIVITY IN X-DIRECTION
705 C PAVG - AVERAGE ATMOSPHERIC PRESSURE (Pa)
706 C VIS - VISCOSITY OF FLUID (Pa-sec)
707 C PERM(I) - PERMEABILITY OF LAYER I (sq.m.)
708 C POR(I) - POROSITY OF LAYER I
709 C
710 DOUBLE PRECISION P(20,200), PO(20,200), SF(20,200),
711 X C(20,200,2), DL(20,200), PC(20), ZC(200), AD(20)
712 DOUBLE PRECISION B(200), G(200), W, HA, BB, CC, DD, D, F, DELTA,
713 X ERROR, E, EN, PAVG, VIS, TIME, QT, HA, PI, TWOPI, TWO, CL, SFL
714 DIMENSION NV(10), ZCS(200), ACS(20), AC(20)
715 DIMENSION NSUB(10), NBSCUB(200), NVSUB(10,10)
716 DIMENSION PSURF(1000), PCAVT(1000), T(1000)
717 CHARACTER*20 PFLAG
718 C
719 COMMON /DAT/ PSURF, PCAVT, DT, ER, PAVG, PFLAG

```

```

720      COMMON/GEOM/NLAYER, NSUB, NV, NVSUB, NBCSUB, RC, ZC, NR, NC, RMIN
721      COMMON/ZAP/JFR(20), NTERMS, NFREE, NPTS
722 C
723      PI=3.14159265D0
724      TWO=2.0D0
725      TWOP= TWO*PI
726      VIS=2.0D-05
727      DELTA=DBLE(DT)
728      ERRDR=DBLE(ER)
729 C
730      IF (KK.GT.1) GO TO 4600
731      IF (PFLAG.EQ.'Y') WRITE(ID2,11000)
732 C
733 C      DECODE PARAMETERS
734      DO 100 J=1, NTERMS
1 735          AD(JFR(J))=DBLE(AC(J))
1 736      100 CONTINUE
737 C
738 C      AD(1) ... PERMEABILITY (DARCY) OF LAYER 1
739 C      AD(2) ... POROSITY OF LAYER 1
740 C      AD(3) ... PERMEABILITY (DARCY) OF LAYER 2
741 C      AD(4) ... POROSITY OF LAYER 2
742 C      ETC.
743 C
744 C
745      MTOP=MAX0(NR,NC)
746 C
747 C      COMPUTE NODAL CONDUCTIVITIES AND STORAGE FACTORS
748 C      INITIALIZE PRESSURE MATRICES
749 C
750      L=1
751      DO 300 J=1,NR
1 752          IF (J.GT.NV(L)) L=L+1
1 753          CL=AD(2*L-1)*9.86923D-13/VIS
1 754          SFL=AD(2*L)/PAVG
1 755          DZ=ZC(MIN0(J+1,NR))-ZC(MAX0(J-1,1))
1 756              DO 200 I=1,NC
2 757                  DR=RC(MIN0(I+1,NC))-RC(MAX0(I-1,1))
2 758                  IF (I.NE.NC) C(I,J,2)=CL*DZ/((RC(I+1)-RC(I))*2.0D0)
2 759                  IF (J.NE.NR) C(I,J,1)=CL*DR/((ZC(J+1)-ZC(J))*2.0D0)
2 760                  SF(I,J)=SFL*DZ*DR/4.0D0
2 761                  PQ(I,J)=PAVG
2 762                  DL(I,J)=0.0D0
2 763      200 P(I,J)=PAVS
1 764 C
1 765 C
1 766 C      SET BOUNDARY NODES
1 767 C
1 768      IF (NBCSUB(J).EQ.2) GO TO 300
1 769      SF(1,J)=SF(1,J)/AD(2*L)
1 770      C(1,J,2)=NBCSUB(J)*2.0D0*C(1,J,2)+0.75D0/0.5D0
1 771      C(1,J,1)=C(1,J,1)*(RMIN**2.0D0)/(8.0D0*9.86923D-13*
1 772      X      AD(2*L-1))
1 773      C(NC,J,2)=0.0D0
1 774      300 CONTINUE
775 C
776      DO 400 I=1,NC
1 777      400 C(I,NR,1)=0.0D0
778 C
779 C
780 C      REFINE NODAL VALUES ON BOUNDARIES OF HORIZONTAL LAYERS
781 C
782      IF (NLAYER.EQ.1) GO TO 600
783      DO 500 L=1,NLAYER-1
1 784          DO 500 I=1,NC
2 785              C(I,NV(L),1)=C(I,NV(L)+1,1)

```

```

2 786          C(I,NV(L),2)=(C(I,NV(L)-1,2)+C(I,NV(L)+1,2))/2.0D0
2 787 500      SF(I,NV(L))=(SF(I,NV(L)-1)+SF(I,NV(L)+1))/2.0D0
788 C
789 C          CONVERT FOR RADIAL FLOW WIDTHS
790 C
791 600      DO 800 J=1,NR
1 792          DO 700 I=2,NC-1
2 793          C(I,J,1)=C(I,J,1)*(RC(I-1)+RC(I+1))*PI
2 794          SF(I,J)=SF(I,J)*(RC(I-1)+RC(I+1))*PI
2 795 700      C(I,J,2)=C(I,J,2)*(RC(I)+RC(I+1))*PI
1 796          C(NC,J,1)=C(NC,J,1)*(RC(NC-1)+0.75D0*(RC(NC)-RC(NC-1)))*TWO
1 797          SF(NC,J)=SF(NC,J)*(RC(NC-1)+0.75D0*(RC(NC)-RC(NC-1)))*TWOPI
1 798          SF(1,J)=SF(1,J)*RC(2)*PI/2.0D0
1 799          C(1,J,1)=C(1,J,1)+0.25D0*RC(2)*PI
1 800 800      C(1,J,2)=C(1,J,2)+0.50D0*RC(2)*PI
801 C
802 C
803          DO 1000 I=2,NC
1 804 1000      SF(I,1)=1.0D25
805 C
806 C          ZERO OUT B AND G ARRAYS
807          DO 1100 J=1,MTOP
1 808          B(J)=0.0D0
1 809          G(J)=0.0D0
1 810 1100      CONTINUE
811 C
812 C          START SIMULATION
813 C
814          TIME=0.0D+00
815          DO 4500 ISTEP=1,NPTS
1 816          TIME=TIME+DELTA
1 817          TLM=SNGL(TIME/3600.0D0)
818 C
1 819 C          SET UPPER BOUNDARY PRESSURE EQUAL TO BAROMETRIC PRESSURE
1 820 C          AT CURRENT TIME
1 821 C
1 822          DO 1300 I=2,NC
2 823          PO(I,1)=DBLE(PSURF(ISTEP))
2 824 1300      P(I,1)=DBLE(PSURF(ISTEP))
1 825 C
1 826 C          PREDICT PRESSURES FOR NEXT TIME INCREMENT
1 827 C
1 828          DO 1500 I=1,NC
2 829          DO 1500 J=1,NR
3 830          IF (SF(I,J).GT.1.0D20) P(I,J)=PO(I,J)
3 831          D=P(I,J)-PO(I,J)
3 832          PO(I,J)=P(I,J)
3 833          F=1.0D0
3 834 1500      P(I,J)=P(I,J)+D*F
1 835 C
1 836 C
1 837 C          REFINE ESTIMATES OF PRESSURES BY IADI METHOD
1 838 C
1 839          ITER=0
1 840 1500      E=0.0D+00
1 841          ITER=ITER+1
1 842 C
1 843 C          COLUMN CALCULATIONS
1 844 C
1 845          DO 3000 II=1,NC
2 846          I=II
2 847          IF (MOD(ISTEP+ITER,2).EQ.1) I=NC-I+1
2 848          DO 2800 J=1,NR
3 849 C
3 850 C          CALCULATE B AND G ARRAYS
3 851 C

```



```

3 852 BB=SF(I,J)/DELTA
3 853 DD=PD(I,J)*SF(I,J)/DELTA
3 854 IF (J-1) 1700,1800,1700
3 855 1700 AA=-C(I,J-1,1)
3 856 BB=BB+C(I,J-1,1)
3 857 1800 IF (J-NR) 1900,2000,1900
3 858 1900 CC=-C(I,J,1)
3 859 BB=BB+C(I,J,1)
3 860 2000 IF (I-1) 2100,2200,2100
3 861 2100 BB=BB+C(I-1,J,2)
3 862 DD=DD+P(I-1,J)*C(I-1,J,2)
3 863 2200 IF (I-NC) 2300,2400,2300
3 864 2300 BB=BB+C(I,J,2)
3 865 DD=DD+P(I+1,J)*C(I,J,2)
3 866 2400 W=BB
3 867 IF (J-1) 2500,2600,2500
3 868 2500 W=W-AA*B(J-1)
3 869 2600 B(J)=CC/W
3 870 G(J)=DD/W
3 871 IF (J-1) 2800,2900,2700
3 872 2700 G(J)=G(J)-AA*G(J-1)/W
3 873 2800 CONTINUE

```

```
2 874 C
```

RE-ESTIMATE PRESSURES

```
2 875 C
```

```
2 876 C
```

```
2 877
```

```
2 878
```

```
2 879
```

```
2 880
```

```
2 881 2900
```

```
2 882
```

```
2 883
```

```
2 884
```

```
2 885
```

```
2 886
```

```
2 887 3000
```

```
1 888 C
```

```
1 889 C
```

```
1 890 C
```

```
1 891
```

```
2 892
```

```
2 893
```

```
2 894
```

```
3 895
```

```
3 896
```

```
3 897
```

```
3 898 3100
```

```
3 899
```

```
3 900 3200
```

```
3 901 3300
```

```
3 902
```

```
3 903 3400
```

```
3 904 3500
```

```
3 905
```

```
3 906 3600
```

```
3 907 3700
```

```
3 908
```

```
3 909 3800
```

```
3 910
```

```
3 911 3900
```

```
3 912 4000
```

```
3 913
```

```
3 914
```

```
3 915 4100
```

```
3 916 4200
```

```
2 917 C
```

```
EN=DABS(P(I,NR)-G(NR))
```

```
E=DMAX1(E,EN)
```

```
P(I,NR)=G(NR)
```

```
N=NR-1
```

```
HA=G(N)-B(N)*P(I,N+1)
```

```
EN=DABS(HA-P(I,N))
```

```
E=DMAX1(E,EN)
```

```
P(I,N)=HA
```

```
N=N-1
```

```
IF (N) 3000,3000,2900
```

```
CONTINUE
```

ROW CALCULATIONS

```
DO 4400 JJ=1,NR
```

```
J=JJ
```

```
IF (MOD(ISTEP+ITER,2).EQ.1) J=NR-J+1
```

```
DD 4200 I=1,NC
```

```
BB=SF(I,J)/DELTA
```

```
DD=PD(I,J)*SF(I,J)/DELTA
```

```
IF (J-1) 3100,3200,3100
```

```
BB=BB+C(I,J-1,1)
```

```
DD=DD+P(I,J-1)*C(I,J-1,1)
```

```
IF (J-NR) 3300,3400,3300
```

```
DD=DD+P(I,J+1)*C(I,J,1)
```

```
BB=BB+C(I,J,1)
```

```
IF (I-1) 3500,3600,3500
```

```
BB=BB+C(I-1,J,2)
```

```
AA=-C(I-1,J,2)
```

```
IF (I-NC) 3700,3800,3700
```

```
BB=BB+C(I,J,2)
```

```
CC=-C(I,J,2)
```

```
W=BB
```

```
IF (I-1) 4000,4000,3900
```

```
W=W-AA*B(I-1)
```

```
B(I)=CC/W
```

```
G(I)=DD/W
```

```
IF (I-1) 4200,4200,4100
```

```
G(I)=G(I)-AA*G(I-1)/W
```

```
CONTINUE
```

```

2  918 C      RE-ESTIMATE PRESSURES
2  919 C
2  920          EN=DABS(P(NC,J)-G(NC))
2  921          E=DMAX1(E,EN)
2  922          P(NC,J)=G(NC)
2  923          N=NC-1
2  924 4300      HA=G(N)-B(N)*P(N+1,J)
2  925          EN=DABS(P(N,J)-HA)
2  926          E=DMAX1(E,EN)
2  927          ES=SNGL(E)
2  928          P(N,J)=HA
2  929          N=N-1
2  930          IF (N) 4400,4400,4300
2  931 4400      CONTINUE
1  932 C
1  933          IF (E.GT.ERROR) GO TO 1600
1  934          IF (ITER.GT.100) WRITE (ID2,12000)
1  935          IF (ITER.GT.100) STOP
1  936          PCAVT(ISTEP)=SNGL(P(1,2))
1  937          IF (PFLAG.EQ.'N') GOTO 4500
1  938          WRITE (ID2,10000) ISTEP,ITER,TLM,PSURF(ISTEP),PCAVT(ISTEP)
1  939 4500      CONTINUE
2  940 C
2  941 C
2  942 4600 FUNCTN=PCAVT(KK)
2  943 4700 RETURN
2  944 10000 FORMAT(1H,3X,2(I3,2X),F10.4,2X,2(F12.6,2X))
2  945 11000 FORMAT(///,3X,' ISTEP ITER TIME (Hrs) PSURF (Pa) PCAV (Pa)',/
2  946 12000 FORMAT(1H,' MORE THAN 100 ITERATIONS IN FUNCTN...STOP ',
2  947          X ' EXECUTION')
2  948          END

```

C)

```

C      ++++++
C      ++++++
C      ++++++ KOBRA: MODULE 2 ++++++
C      ++++++
C      ++++++
C      UNLESS OTHERWISE STATED, ALL OF THE FOLLOWING SUBROUTINES
C      CAN BE FOUND IN:
C      "DATA REDUCTION AND ERROR ANALYSIS FOR THE PHYSICAL
C      SCIENCES". P.R. BEVINGTON. MCGRAW-HILL. 1969
C
1      SUBROUTINE CURFIT(X,Y,SIGMAY,NPTS,NTERMS,MODE,A,DELTA,
2      X VAR,FLAMDA,YFIT,CHISQR,NVAR)
3      DOUBLE PRECISION ARRAY
4      DIMENSION X(1),Y(1),SIGMAY(1),A(1),DELTA(1),YFIT(1)
5      DIMENSION WEIGHT(1000),ALPHA(20,20),BETA(20),DERIV(20,1000).
6      X ARRAY(20,20),B(20),VAR(20,20)
7      CHARACTER*20 QFLAG
8
9      OPERATIONAL MODIFICATIONS TO BEVINGTON ARE INDICATED BY
10     ASTERISKS
11     *****
12     COMMON/STUFF/HSTART
13     COMMON/DER/LPTS
14     COMMON/STUFF2/QFLAG
15     *****
16     ID1=0
17     ID2=0
18
19     PURPOSE:
20     MAKE A LEAST-SQUARES FIT TO A NON-LINEAR FUNCTION
21
22     USAGE:
23     CALL CURFIT(X,Y,SIGMAY,NPTS,NTERMS,MODE,A,DELTA,
24     X SIGMA,FLAMDA,YFIT,CHISQR)
25
26     DESCRIPTION OF PARAMETERS
27     X      - ARRAY OF DATA POINTS FOR INDEPENDENT VARIABLE
28     Y      - ARRAY OF DATA POINTS FOR DEPENDENT VARIABLE
29     SIGMAY - ARRAY OF STANDARD DEVIATIONS FOR Y DATA POINTS
30     NPTS   - NUMBER OF PAIRS OF DATA POINTS
31     NTERMS - NUMBER OF FREE PARAMETERS
32     MODE   - DETERMINES METHOD OF WEIGHTING LEAST-SQUARES FIT
33             +1 (INSTRUMENTAL) WEIGHT(I)=1./SIGMAY(I)**2
34             0 (NO WEIGHTING) WEIGHT(I)=1.
35             -1 (STATISTICAL) WEIGHT(I)=1./Y(I)
36     A      - ARRAY OF PARAMETERS
37     DELTA  - ARRAY OF INCREMENTS FOR PARAMETERS A
38     VAR    - VARIANCE-COVARIANCE MATRIX FOR PARAMETERS A
39     FLAMDA - PROPORTION OF GRADIENT SEARCH INCLUDED
40     YFIT   - ARRAY OF CALCULATED VALUES OF Y
41     CHISQR - REDUCED CHI SQUARE FOR FIT
42
43     SUBROUTINES AND FUNCTION SUBPROGRAMS REQUIRED
44     FUNCTN(X,I,A)
45     EVALUATES THE FITTING FUNCTION FOR THE ITH TERM
46     FCHISQ(Y,SIGMAY,NPTS,NFREE,MODE,YFIT)
47     EVALUATES REDUCED CHI SQUARE FOR FIT TO DATA
48     FDERIV(X,I,A,DELTA,NTERMS,DERIV)
49     EVALUATES THE DERIVATIVES OF THE FITTING FUNCTION
50     FOR THE ITH TERM WITH RESPECT TO EACH PARAMETER
51     MATINV(ARRAY,NTERMS,DET)

```

```

52 C      INVERTS A SYMMETRIC TWO-DIMENSIONAL MATRIX OF DEGREE NTERMS
53 C      AND CALCULATES ITS DETERMINANT
54 C
55 C      COMMENTS:
56 C      DIMENSION STATEMENT VALID FOR NTERMS UP TO 10
57 C      SET FLAMDA=0.001 AT BEGINNING OF SEARCH
58 C
59 C
60      NFREE= NPTS-NTERMS
61      IF (NFREE) 200,200,300
62 200 CHISQ=0.
63      GO TO 3300
64 C
65 C      EVALUATE WEIGHTS
66 C
67 300      DO 1000 I=1,NPTS
68          IF (MODE) 500,800,900
69 500          IF (Y(I)) 700,800,600
70 600          WEIGHT(I)=1./Y(I)
71          GO TO 1000
72 700          WEIGHT(I)=1./(-Y(I))
73          GO TO 1000
74 800          WEIGHT(I)=1.
75          GO TO 1000
76 900          WEIGHT(I)=1./SIGMAY(I)**2
77 1000         CONTINUE
78 C
79 C
80 C      EVALUATE ALPHA AND BETA MATRICES
81 C
82 C      *****
83      DO 1200 J=1,NTERMS
84          BETA(J)=0.
85          DO 1200 K=1,J
86 1200         ALPHA(J,K)=0.
87          IF(QFLAG.EQ.'Y') WRITE(ID2,*) ' COMPUTING INITIAL FORWARD ',
88          X 'MODEL'
89          DO 1300 K=1,NPTS
90 1300         YFIT(K)=FUNCTN(X,K,A)
91 C
92 C      EVALUATE CHISQUARE AT STARTING POINT
93 C
94      CHISQ1=FCHISQ(Y,SIGMAY,NPTS,NFREE,MODE,YFIT)
95      IF(QFLAG.EQ.'Y') WRITE(ID2,*) ' INITIAL REDUCED CHI-SQUARE IS ',
96      X CHISQ1
97 C
98 C
99      IF(QFLAG.EQ.'Y') WRITE(ID2,*) ' ENTERING FDERIV FROM CURFIT'
100      CALL FDERIV(X,1,A,DELTAA,NTERMS,DERIV)
101      IF(QFLAG.EQ.'Y') WRITE(ID2,*) ' CALCULATING ALPHA AND BETA ',
102      X 'ARRAYS'
103      DO 1600 I=MSTART,NPTS
104          DO 1500 J=1,NTERMS
105              BETA(J)=BETA(J)+WEIGHT(I)*(Y(I)-YFIT(I))*DERIV(J,I)
106              DO 1500 K=1,J
107 1500              ALPHA(J,K)=ALPHA(J,K)+WEIGHT(I)*DERIV(J,I)
108              X *DERIV(K,I)
109 1600          CONTINUE
110          DO 1800 J=1,NTERMS
111              DO 1800 K=1,J
112 1800              ALPHA(K,J)=ALPHA(J,K)
113 C      *****
114 C
115 C
116 C      INVERT MODIFIED CURVATURE MATRIX TO FIND NEW PARAMETERS
117 C

```

```

118 2000      DO 2200 J=1, NTERMS
1 119          DO 2100 K=1, NTERMS
2 120          ARRAY(J,K)=ALPHA(J,K)/SQRT(ALPHA(J,J)*ALPHA(K,K))
2 121 2100      CONTINUE
1 122 2200      ARRAY(J,J)=1.+FLAMDA
123          IF(QFLAG.EQ.'Y') WRITE(ID2,*) ' INVERTING MATRIX "ARRAY"'
124          CALL MATINV(ARRAY, NTERMS, DET)
125          DO 2400 J=1, NVAR
1 126          B(J)=A(J)
1 127 2400      CONTINUE
128          IF(QFLAG.EQ.'Y') WRITE(ID2,*) ' UPDATING FREE PARAMETERS'
129          DO 2500 J=1, NTERMS
1 130          DO 2500 K=1, NTERMS
2 131          VAR(J,K)=ARRAY(J,K)/(SQRT(ALPHA(J,J)*ALPHA(K,K)))
2 132          B(J)=B(J)+PETA(K)*VAR(J,K)
2 133          IF(QFLAG.EQ.'Y') WRITE(ID2,5000) J, B(J)
2 134 2500      CONTINUE
135 C
136 C      IF CHI SQUARE INCREASED, INCREASE FLAMDA AND TRY AGAIN
137 C
138          IF(QFLAG.EQ.'Y') WRITE(ID2,*) ' COMPUTE NEW FORWARD MODEL'
139          DO 2700 I=1, NPTS
1 140 2700      YFIT(I)=FUNCTN(X,I,B)
141          CHISQR=FCHISQ(Y, SIGMA, NPTS, NFREE, MODE, YFIT)
142          IF(QFLAG.EQ.'Y') WRITE(ID2,*) ' REDUCED CHI-SQUARE IS ', CHISQR
143          IF (CHISQ1-CHISQR) 2900, 3000, 3000
144 2900      FLAMDA=10.0*FLAMDA
145          IF(QFLAG.EQ.'Y') WRITE(ID2,*) ' CHI-SQUARE INCREASED...',
146          X ' INCREASE LAMDA AND TRY AGAIN'
147          GO TO 2000
148 C
149 C      EVALUATE PARAMETERS AND VARIANCE-COVARIANCE MATRIX
150 C
151 3000      DO 3200 J=1, NTERMS
1 152          DO 3100 K=1, NTERMS
2 153          VAR(J,K)=CHISQR*VAR(J,K)
1 154 3200      A(J)=B(J)
155          FLAMDA=FLAMDA/10.0
156 3300      RETURN
157 5000      FORMAT(1H, 'NEW VALUE OF PARAMETER ', I2, ' IS ', E13.5)
158      END

159 C
160 C      SUBROUTINE GRIDLS
161 C
162 C      PURPOSE
163 C          MAKE A GRID-SEARCH LEAST-SQUARES FIT TO DATA WITH A SPECIFIED
164 C          FUNCTION WHICH IS NOT LINEAR IN COEFFICIENTS
165 C
166 C      USAGE
167 C          CALL GRIDLS(X, Y, SIGMA, NPTS, NTERMS, MODE, A, DELTA, SIGMA,
168 C          YFIT, CHISQR)
169 C
170 C      DESCRIPTION OF PARAMETERS
171 C          X      - ARRAY OF DATA POINTS FOR INDEPENDENT VARIABLE
172 C          Y      - ARRAY OF DATA POINTS FOR DEPENDENT VARIABLE
173 C          SIGMA   - ARRAY OF STANDARD DEVIATIONS FOR Y DATA POINTS
174 C          NPTS    - NUMBER OF PAIRS OF DATA POINTS
175 C          NTERMS  - NUMBER OF PARAMETERS
176 C          MODE    - DETERMINES METHOD OF WEIGHTING LEAST-SQUARES FIT
177 C                   +1 (INSTRUMENTAL) WEIGHT(I)=1.0/SIGMA(I)**2
178 C                   0 (NO WEIGHTING) WEIGHT(I)=1.0
179 C                   -1 (STATISTICAL) WEIGHT(I)=1.0/Y(I)
180 C          A      - ARRAY OF PARAMETERS
181 C          DELTA   - ARRAY OF INCREMENTS FOR PARAMETERS A
182 C          SIGMA   - ARRAY OF STANDARD DEVIATIONS FOR PARAMETERS A

```

```

183 C      YFIT - ARRAY OF CALCULATED VALUES OF Y
184 C      CHISQR - REDUCED CHI SQUARE FOR FIT
185 C
186 C      SUBROUTINES AND FUNCTION SUBPROGRAMS REQUIRED
187 C      FUNCTN(X,I,A)
188 C      EVALUATES THE FITTING FUNCTION FOR THE ITH TERM
189 C      FCHISQ(Y,SIGMA,Y,NPTS,NFREE,MODE,YFIT)
190 C      EVALUATES REDUCED CHI SQUARE FOR FIT TO DATA
191 C
192 C      COMMENTS
193 C      DELTAA VALUES ARE MODIFIED BY THE PROGRAM
194 C
195 C      SUBROUTINE GRIDL (X,Y,SIGMA,Y,NPTS,NTERMS,MODE,A,DELTAA,
196 C      XSIGMA,YFIT,CHISQR)
197 C
198 C      DIMENSION X(1),Y(1),SIGMA(1),A(1),DELTAA(1),SIGMAA(1),YFIT(1)
199 C      CHARACTER*20 QFLAG
200 C
201 C      OPERATIONAL MODIFICATIONS TO BEVINGTON ARE INDICATED BY
202 C      ASTERISKS
203 C      *****
204 C      COMMON/STUFF/MSTART
205 C      COMMON/STUFF2/QFLAG
206 C      *****
207 C      ID1=0
208 C      ID2=0
209 C
210 C      NFREE=NPTS-NTERMS
211 C      FREE=NFREE
212 C      CHISQR=0.0
213 C      IF (NFREE) 2600,2600,200
214 C      200 DO 2100 J=1,NTERMS
1 215 IF (QFLAG.EQ.'Y') WRITE(ID2,5000) J
1 216 C
1 217 C      EVALUATE CHI SQUARE AT FIRST TWO SEARCH POINTS
1 218 C
1 219 C
1 220 IF (QFLAG.EQ.'Y') WRITE(ID2,*) ' COMPUTE INITIAL FORWARD MODEL'
1 221 DO 400 I=1,NPTS
2 222 400 YFIT(I)=FUNCTN(X,I,A)
1 223 CHISQ1=FCHISQ(Y,SIGMA,Y,NPTS,NFREE,MODE,YFIT)
1 224 IF (QFLAG.EQ.'Y') WRITE(ID2,*) ' INITIAL REDUCED CHI-SQUARE IS ',
1 225 X CHISQ1
1 226 FN=0.0
1 227 DELTA=DELTAA(J)
1 228 IF (QFLAG.EQ.'Y') WRITE(ID2,*) ' INCREMENT PARAMETER'
1 229 500 A(J)=A(J)+DELTA
1 230 IF (QFLAG.EQ.'Y') WRITE(ID2,5010) J,A(J)
1 231 IF (QFLAG.EQ.'Y') WRITE(ID2,*) ' COMPUTING FORWARD MODEL'
1 232 DO 700 I=1,NPTS
2 233 700 YFIT(I)=FUNCTN(X,I,A)
1 234 CHISQ2=FCHISQ(Y,SIGMA,Y,NPTS,NFREE,MODE,YFIT)
1 235 IF (QFLAG.EQ.'Y') WRITE(ID2,*) ' REDUCED CHI-SQUARE IS NOW ',CHISQ2
1 236 IF (CHISQ1-CHISQ2) 1000,600,1300
1 237 C
1 238 C      REVERSE DIRECTION OF SEARCH IF CHI SQUARE IS INCREASING
1 239 C
1 240 C
1 241 1000 DELTA=-DELTA
1 242 IF (QFLAG.EQ.'Y') WRITE(ID2,*) ' CHI-SQUARE IS INCREASING...',
1 243 X ' REVERSE DIRECTION OF SEARCH'
1 244 A(J)=A(J)+DELTA
1 245 IF (QFLAG.EQ.'Y') WRITE(ID2,5010) J,A(J)
1 246 IF (QFLAG.EQ.'Y') WRITE(ID2,*) ' COMPUTING FORWARD MODEL'
1 247 DO 1100 I=1,NPTS
2 248 1100 YFIT(I)=FUNCTN(X,I,A)

```

```

1 249          SAVE=CHISQ1
1 250          CHISQ1=CHISQ2
1 251          CHISQ2=SAVE
1 252 C
1 253 C      INCREMENT A(J) UNTIL CHI SQUARE INCREASES
1 254 C
1 255 C
1 256 1300      FN=FN+1.0
1 257          IF(QFLAG.EQ.'Y') WRITE(ID2,*) ' INCREMENT PARAMETER'
1 258          A(J)=A(J)+DELTA
1 259          IF(QFLAG.EQ.'Y') WRITE(ID2,5010) J,A(J)
1 260          IF(QFLAG.EQ.'Y') WRITE(ID2,*) ' COMPUTING FORWARD MODEL'
1 261          DO 1400 I=1,NPTS
2 262 1400      YFIT(I)=FUNCTN(X,I,A)
1 263          CHISQ3=FCHISQ(Y,SIGMAY,NPTS,NFREE,MODE,YFIT)
1 264          IF(QFLAG.EQ.'Y') WRITE(ID2,*) ' REDUCED CHI-SQUARE IS NOW ',
1 265          X CHISQ3
1 266          IF (CHISQ3-CHISQ2) 1600,1700,1700
1 267 1600      CHISQ1=CHISQ2
1 268          CHISQ2=CHISQ3
1 269          GO TO 1300
1 270 C
1 271 C      FIND MINIMUM OF PARABOLA DEFINED BY LAST THREE POINTS
1 272 C
1 273 C
1 274 1700      DELTA=DELTA*(1./(1.+(CHISQ1-CHISQ2)/(CHISQ3-CHISQ2)))+0.5)
1 275          IF(QFLAG.EQ.'Y') WRITE(ID2,*) ' UPDATE PARAMETER BASED ON LAST ',
1 276          X ' THREE POINTS'
1 277          A(J)=A(J)-DELTA
1 278          IF(QFLAG.EQ.'Y') WRITE(ID2,5010, J,A(J))
1 279          SIGMAA(J)=DELTAA(J)*SQRT(2.0/(FREE*(CHISQ3-2.0*CHISQ2+CHISQ1)
1 280          X ))
1 281          DELTAA(J)=DELTAA(J)*FN/3.0
1 282          IF(QFLAG.EQ.'Y') WRITE(ID2,5020) J
1 283 2100      CONTINUE
284 C
285 C      EVALUATE FIT AND CHI SQUARE FOR FINAL PARAMETERS
286 C
287          IF(QFLAG.EQ.'Y') WRITE(ID2,*) ' EVALUATE FINAL FORWARD PROBLEM'
288          DO 2300 I=1,NPTS
1 289 2300      YFIT(I)=FUNCTN(X,I,A)
290          CHISQR=FCHISQ(Y,SIGMAY,NPTS,NFREE,MODE,YFIT)
291          IF(QFLAG.EQ.'Y') WRITE(ID2,*) ' FINAL REDUCED CHI-SQUARE IS ',
292          X CHISQR
293          DO 2500 K=1,NTERMS
1 294 2500      SIGMAA(K)=SQRT(CHISQR)*SIGMAA(K)
295 2600 RETURN
296 5000 FORMAT(1H ,///,' BEGIN VARYING PARAMETER ',I2,' IN GRID SEARCH')
297 5010 FORMAT(1H , ' CURRENT VALUE OF PARAMETER ',I2,' IS ',E12.4)
298 5020 FORMAT(1H , ' END OF PARAMETER ',I2,' VARIATION')
299      END

300      SUBROUTINE FDERIV(X,I,A,DELTAA,NTERMS,DERIV)
301      DIMENSION X(1000),A(20),DELTAA(20),DERIV(20,1000)
302 C      *****
303      DIMENSION YFIT1(1000),YFIT2(1000)
304      CHARACTER*20 QFLAG
305 C      *****
306 C
307 C      OPERATIONAL MODIFICATIONS TO BEVINGTON ARE INDICATED BY
308 C      ASTERISKS
309 C      *****
310      COMMON/DER/LPTS
311      COMMON/STUFF2/QFLAG
312 C      *****
313 C

```

```

314 C      PURPOSE:
315 C      EVALUATE DERIVATIVES OF FUNCTION FOR LEAST-SQUARES
316 C      SEARCH FOR ARBITRARY FUNCTION GIVEN BY FUNCTN
317 C
318 C      USAGE:
319 C      CALL FDERIV(X,I,A,DELTAA,NTERMS,DERIV)
320 C
321 C      DESCRIPTION OF PARAMETERS
322 C      X      - ARRAY OF DATA POINTS FOR INDEPENDENT VARIABLE
323 C      I      - INDEX OF DATA POINTS
324 C      A      - ARRAY OF PARAMETERS
325 C      DELTAA - ARRAY OF PARAMETER INCREMENTS
326 C      NTERMS - NUMBER OF PARAMETERS
327 C      DERIV  - DERIVATIVES OF FUNCTION
328 C
329 C      SUBROUTINE AND FUNCTION SUBPROGRAMS REQUIRED
330 C      FUNCTN(X,I,A)
331 C      EVALUATES THE FITTING FUNCTION FOR THE ITH TERM
332 C      *****
333 C      DO 700 J=1,NTERMS
1 334 C
1 335 C      IF(QFLAG.EQ.'Y') WRITE(ID2,5000) J
1 336 C      DELTA=DELTAA(J)
1 337 C      AJ=A(J)
1 338 C      IF(QFLAG.EQ.'Y') WRITE(ID2,*) ' INCREMENT PARAMETER'
1 339 C      A(J)=AJ+DELTA
1 340 C
1 341 C      IF(QFLAG.EQ.'Y') WRITE(ID2,*) ' COMPUTE FORWARD MODEL'
1 342 C      DO 200 K=1,LPTS
2 343 C      YFIT1(K)=FUNCTN(X,K,A)
2 344 C      200 CONTINUE
1 345 C
1 346 C      IF(QFLAG.EQ.'Y') WRITE(ID2,*) ' DECREMENT PARAMETER'
1 347 C      A(J)=AJ-DELTA
1 348 C
1 349 C      IF(QFLAG.EQ.'Y') WRITE(ID2,*) ' COMPUTE FORWARD MODEL'
1 350 C      DO 300 K=1,LPTS
2 351 C      YFIT2(K)=FUNCTN(X,K,A)
2 352 C      300 CONTINUE
1 353 C
1 354 C      400 A(J)=AJ
1 355 C
1 356 C      DO 600 K=1,LPTS
2 357 C      DERIV(J,K)=(YFIT1(K)-YFIT2(K))/(2.0*DELTAA(J))
2 358 C      600 CONTINUE
1 359 C      700 CONTINUE
1 360 C      5000 FORMAT(1H,' PARAMETER ',I2,' IN FDERIV')
361 C      *****
362 C      RETURN
363 C      END
364 C
365 C      FUNCTION FCHISQ(Y,SIGMA,NPTS,NFREE,MODE,YFIT)
366 C      DOUBLE PRECISION CHISQ,WEIGHT
367 C
368 C      OPERATIONAL MODIFICATIONS TO BEVINGTON ARE INDICATED BY
369 C      ASTERISKS
370 C      *****
371 C      COMMON/STUFF/ASTART
372 C      *****
373 C      DIMENSION Y(1),SIGMA(1),YFIT(1)
374 C
375 C      PURPOSE
376 C      EVALUATE REDUCED CHI SQUARE FOR FIT TO DATA
377 C      FCHISQ=SUM((Y-YFIT)**2/SIGMA**2)/NFREE
378 C
379 C      USAGE

```



```

379 C      RESULT=FCHISQ(Y,SIGMAY,NPTS,NFREE,MODE,YFIT)
380 C
381 C      DESCRIPTION OF PARAMETERS
382 C      Y      - ARRAY OF DATA POINTS
383 C      SIGMAY - ARRAY OF STANDARD DEVIATIONS FOR DATA POINTS
384 C      NPTS   - NUMBER OF DATA POINTS
385 C      NFREE   - NUMBER OF DEGREES OF FREEDOM
386 C      MODE    - DETERMINES METHOD OF WEIGHTING LEAST-SQUARES FIT
387 C      - +1 (INSTRUMENTAL) WEIGHT(I)=1./SIGMAY(I)**2
388 C      - 0 (NO WEIGHTING) WEIGHT(I)=1.
389 C      - -1 (STATISTICAL) WEIGHT(I)=1./Y(I)
390 C
391 C      YFIT    - ARRAY OF CALCULATED VALUES OF Y
392 C
393 C
394 C
395 C      CHISQ=0.
396 C      IF (NFREE) 300,300,400
397 300 FCHISQ=0.
398 C      GO TO 1400
399 C
400 C      ACCUMULATE CHI SQUARE
401 C
402 C      *****
403 400 DO 1100 I=MSTART,NPTS
404 C      *****
1 405 IF (MODE) 600,900,1000
1 406 600 IF (Y(I)) 800,900,700
1 407 700 WEIGHT=1./Y(I)
1 408 GO TO 1100
1 409 800 WEIGHT=1./(-Y(I))
1 410 GO TO 1100
1 411 900 WEIGHT=1.
1 412 GO TO 1100
1 413 1000 WEIGHT=1./SIGMAY(I)**2
1 414 1100 CHISQ=CHISQ+WEIGHT*(Y(I)-YFIT(I))**2
415 C
416 C      DIVIDE BY NUMBER OF DEGREES OF FREEDOM
417 C
418 C      FREE=NFREE
419 C      FCHISQ=CHISQ/FREE
420 1400 RETURN
421 C      END

422 C      SUBROUTINE RAN(R)
423 C      FROM PEERLESS ENGINEERING SERVICE
424 C      PUBLISHED AND COPYRIGHTED BY JOHN WILEY AND SONS
425 C
426 C      THIS SUBROUTINE GENERATES A SEQUENCE OF NUMBERS WHICH ARE
427 C      RANDOMLY AND UNIFORMLY DISTRIBUTED OVER THE UNIT INTERVAL
428 C
429 C      BB=1
430 C      P1=R*317.
431 C      R=AMOD(P1,BB)
432 C      RETURN
433 C      END

434 C      FUNCTION PCORRE(R,NPTS,NTERMS)
435 C
436 C      PURPOSE
437 C      EVALUATE PROBABILITY FOR NO CORRELATION BETWEEN TWO VARIABLES
438 C
439 C      DESCRIPTION OF PARAMETERS
440 C      R      - LINEAR CORRELATION COEFFICIENT
441 C      NPTS   - NUMBER OF DATA POINTS
442 C      NTERMS - NUMBER OF MODEL PARAMETERS

```

```

443 C
444 C      SUBROUTINES AND FUNCTION SUBPROGRAMS REQUIRED
445 C      GAMMA(X)
446 C      CALCULATES GAMMA FUNCTION FOR INTEGERS AND HALF-INTEGERS
447 C
448 C      DOUBLE PRECISION R2, TERM, SUM, FI, FNUM, DENOM
449 C
450 C      EVALUATE NUMBER OF DEGREES OF FREEDOM
451 C      NFREE=NPTS-NTERMS
452 C      IF (NFREE) 200, 200, 300
453 200 PCORRE=0.0
454 C      GO TO 1800
455 300 R2=R**2
456 C      IF (1.0-R2) 200, 200, 400
457 400 NEVEN=2*(NFREE/2)
458 C      IF (NFREE-NEVEN) 500, 500, 1000
459 C
460 C      NUMBER OF DEGREES OF FREEDOM IS EVEN
461 500 IMAX=(NFREE-2)/2
462 C      FREE=NFREE
463 C      TERM=ABS(R)
464 C      SUM=TERM
465 C      IF (IMAX) 1800, 700, 800
466 700 PCORRE=1.0-TERM
467 C      GO TO 1800
468 800 DO 900 I=1, IMAX
1 469 C      FI=I
1 470 C      FNUM=IMAX-I+1
1 471 C      DENOM=2*I+1
1 472 C      TERM=-TERM*R2*FNUM/FI
1 473 900 SUM=SUM+TERM/DENOM
474 C      PCORRE=1.128379167*(GAMMA((FREE+1.0)/2.0)/GAMMA(FREE/2.0))
475 C      PCORRE=1.0-PCORRE*SUM
476 C      GO TO 1800
477 C
478 C      NUMBER OF DEGREES OF FREEDOM IS ODD
479 C
480 1000 IMAX=(NFREE-3)/2
481 C      TERM=ABS(R)*DSQRT(1.0-R2)
482 C      SUM=DATAN(R2/TERM)
483 C      IF (IMAX) 1700, 1300, 1400
484 1300 SUM=SUM+TERM
485 C      GO TO 1700
486 1400 SUM=SUM+TERM
487 C      DO 1600 I=1, IMAX
1 488 C      FNUM=2*I
1 489 C      DENOM=2*I+1
1 490 C      TERM=TERM*(1.0-R2)*FNUM/DENOM
1 491 1600 SUM=SUM+TERM
492 1700 PCORRE=1.0-0.6366197724*SUM
493 1800 RETURN
494 C      END
495 C      FUNCTION PCHISQ (CHISQ, NFREE)
496 C
497 C      PURPOSE
498 C      EVALUATE PROBABILITY FOR EXCEEDING CHI SQUARE
499 C
500 C      USAGE
501 C      RESULT = PCHISQ (CHISQ, NFREE)
502 C
503 C      DESCRIPTION OF PARAMETERS
504 C      CHISQ - COMPARISON VALUE OF REDUCED CHI SQUARE
505 C      NFREE - NUMBER OF DEGREES OF FREEDOM
506 C
507 C      SUBROUTINES AND FUNCTION SUBPROGRAMS REQUIRED

```

```

508 C      GAMMA(X)
509 C      CALCULATES GAMMA FUNCTION FOR INTEGERS AND HALF-
510 C      INTEGERS
511 C
512 C      COMMENTS
513 C      CALCULATION IS APPROXIMATE FOR NFREE ODD AND CHI SQUARE
514 C      GREATER THAN 50
515 C
516 C      DOUBLE PRECISION Z,TERM,SUM,FACT
517 C      IF (NFREE) 200,200,300
518 200 PCHISQ=0.
519 C      GO TO 1500
520 300 FREE=NFREE
521 C      Z=CHISQR*FREE/2.
522 C      NEVEN=2*(NFREE/2)
523 C      IF (NFREE-NEVEN) 400,400,800
524 C
525 C      NUMBER OF DEGREES OF FREEDOM IS EVEN
526 C
527 400 IMAX=NFREE/2
528 C      TERM=1.000
529 C      SUM=0.000
530 C      DO 600 I=1,IMAX
1 531 C      FI=I
1 532 C      SUM=SUM+TERM
1 533 C      IF (ABS(TERM).GT.1.0E20) GOTO 650
1 534 600 TERM=TERM*Z/FI
535 650 FACT=0.000
536 C      IF (DABS(Z).GT.70.000) GO TO 700
537 C      FACT=DEXP(-Z)
538 700 PCHISQ=SUM*FACT
539 C      GO TO 1500
540 C
541 C      NUMBER OF DEGREES OF FREEDOM IS ODD
542 C
543 800 IF (Z-25) 1000,1000,900
544 900 Z=CHISQR*(FREE-1.)/2.
545 C      GO TO 400
546 1000 PWR=FREE/2.
547 C      TERM=1.
548 C      SUM=TERM/PWR
549 C      DO 1300 I=1,1000
1 550 C      FI=I
1 551 C      TERM=-TERM*Z/FI
1 552 C      SUM=SUM+TERM/(PWR+FI)
1 553 C      IF (DABS(TERM/SUM)-0.00001) 1400,1400,1300
1 554 1300 CONTINUE
555 1400 PCHISQ=1.-(Z*PWR)*SUM/GAMMA(PWR)
556 1500 RETURN
557 C      END

558 C      FUNCTION GAMMA(X)
559 C
560 C      PURPOSE
561 C      CALCULATE THE GAMMA FUNCTION FOR INTEGERS AND HALF-
562 C      INTEGERS
563 C
564 C      USAGE
565 C      RESULT=GAMMA(X)
566 C
567 C      DESCRIPTION OF PARAMETERS
568 C      X - INTEGER OR HALF-INTEGER
569 C
570 C      SUBROUTINES AND FUNCTION SUBPROGRAMS REQUIRED
571 C      FACTOR(N)
572 C      CALCULATES N FACTORIAL FOR INTEGERS

```

```

573 C
574     DOUBLE PRECISION PROD,SUM,FI
575 C
576 C     INTEGERIZE ARGUMENT
577 C
578     N=X-0.25
579     XN=N
580     IF (X-XN-0.75) 400,400,300
581 C
582 C     ARGUMENT IS INTEGER
583 C
584     300 GAMMA=FACTOR(N)
585     GO TO 1200
586 C
587 C     ARGUMENT IS HALF-INTEGER
588 C
589     400 PROD=1.77245385
590     IF (N) 800,800,500
591     500 IF (N-10) 600,600,900
592     600     DO 700 I=1,N
1 593         FI=I
1 594     700     PROD=PROD*(FI-0.5)
595     800 GAMMA=PROD
596     GO TO 1200
597     900 SUM=0.0
598     DO 1000 I=11,N
1 599         FI=I
1 600     1000     SUM=SUM+DLOG(FI-0.5)
601     IF (SUM.GE.71.78) SUM=71.78
602     GAMMA=PROD*639383.8623*DEXP(SUM)
603     1200 RETURN
604     END

605     FUNCTION FACTOR(N)
606 C
607 C     PURPOSE
608 C         CALCULATES FACTORIAL FUNCTION FOR INTEGERS
609 C
610 C     DESCRIPTION OF PARAMETERS
611 C         N - INTEGER ARGUMENT
612 C
613     DOUBLE PRECISION FI,SUM
614     FACTOR=1.
615     IF (N-1) 800,800,200
616     200 IF (N-10) 300,300,500
617 C
618 C         N LESS THAN 11
619 C
620     300     DO 400 I=2,N
1 621         FI=I
1 622     400     FACTOR=FACTOR*FI
623     GO TO 800
624 C
625 C         N GREATER THAN 10
626 C
627     500 SUM=0.0
628     DO 600 I=11,N
1 629         FI=I
1 630     600     SUM=SUM+DLOG(FI)
631     FACTOR=3628800.*DEXP(SUM)
632     800 RETURN
633     END

634     SUBROUTINE MATINV(ARRAY,NORDER,DET)
635 C
636 C     PURPOSE

```

```

637 C      INVERT A SYMMETRIC MATRIX AND CALCULATE ITS DETERMINANT
638 C
639 C      DESCRIPTION OF PARAMETERS
640 C      ARRAY      - INPUT MATRIX WHICH IS REPLACED BY ITS INVERSE
641 C      NORDER     - DEGREE OF MATRIX (ORDER OF DETERMINANT)
642 C      DET        - DETERMINANT OF INPUT MATRIX
643 C
644 C      COMMENTS
645 C      DIMENSION STATEMENT VALID FOR NORDER UP TO 10
646 C
647 C      DOUBLE PRECISION ARRAY,AMAX,SAVE
648 C      DIMENSION ARRAY(20,20),IK(20),JK(20)
649 C      DET=1.
650 C      DO 2500 K=1,NORDER
1 651 C
1 652 C      FIND LARGEST ELEMENT ARRAY(I,J) IN REST OF MATRIX
1 653 C
1 654 C      AMAX=0.
1 655 C      DO 600 I=K,NORDER
2 656 C      DO 600 J=K,NORDER
3 657 C      IF (DABS(AMAX)-DABS(ARRAY(I,J))) 500,500,600
3 658 C      AMAX=ARRAY(I,J)
3 659 C      IK(K)=I
3 660 C      JK(K)=J
3 661 C      CONTINUE
1 662 C
1 663 C      INTERCHANGE ROWS AND COLUMNS TO PUT AMAX IN ARRAY(K,K)
1 664 C
1 665 C      IF (AMAX) 900,800,900
1 666 C      DET=0.
1 667 C      GO TO 3300
1 668 C      I=IK(K)
1 669 C      IF (I-K) 300,1200,1000
1 670 C      DO 1100 J=1,NORDER
2 671 C      SAVE=ARRAY(K,J)
2 672 C      ARRAY(K,J)=ARRAY(I,J)
2 673 C      ARRAY(I,J)=SAVE
1 674 C      J=JK(K)
1 675 C      IF (J-K) 300,1500,1300
1 676 C      DO 1400 I=1,NORDER
2 677 C      SAVE=ARRAY(I,K)
2 678 C      ARRAY(I,K)=ARRAY(I,J)
2 679 C      ARRAY(I,J)=SAVE
1 680 C
1 681 C      ACCUMULATE ELEMENTS OF INVERSE MATRIX
1 682 C
1 683 C      DO 1700 I=1,NORDER
2 684 C      IF (I-K) 1600,1700,1600
2 685 C      ARRAY(I,K)=-ARRAY(I,K)/AMAX
2 686 C      CONTINUE
1 687 C      DO 2100 I=1,NORDER
2 688 C      DO 2100 J=1,NORDER
3 689 C      IF (I-K) 1900,2100,1900
3 690 C      IF (J-K) 2000,2100,2000
3 691 C      ARRAY(I,J)=ARRAY(I,J)+ARRAY(I,K)*ARRAY(K,J)
3 692 C      CONTINUE
1 693 C      DO 2400 J=1,NORDER
2 694 C      IF (J-K) 2300,2400,2300
2 695 C      ARRAY(K,J)=ARRAY(K,J)/AMAX
2 696 C      CONTINUE
1 697 C      ARRAY(K,K)=1./AMAX
1 698 C      DET=DET*AMAX
699 C
700 C      RESTORE ORDERING OF MATRIX
701 C
702 C      DO 3200 L=1,NORDER

```

```

1  703          K=NORDER-L+1
1  704          J=IK(K)
1  705          IF (J-K) 2900,2900,2700
1  706 2700      DO 2800 I=1,NORDER
2  707          SAVE=ARRAY(I,K)
2  708          ARRAY(I,K)=-ARRAY(I,J)
2  709 2800      ARRAY(I,J)=SAVE
1  710 2900      I=JK(K)
1  711          IF (I-K) 3200,3200,3000
1  712 3000      DO 3100 J=1,NORDER
2  713          SAVE=ARRAY(K,J)
2  714          ARRAY(K,J)=-ARRAY(I,J)
2  715 3100      ARRAY(I,J)=SAVE
1  716 3200      CONTINUE
717 3300 RETURN
718          END

```

COMPUTER PROGRAM: ONEDEE

ONEDEE solves the forward and inverse problems for the 1-D vertically averaged flow model. This model has been shown to be an excellent approximation to the 2-D (vertical and radial) flow model for the case of a homogeneous and isotropic half-space. Method of solution is semi-analytical.

```

D)
C *****
C ++++++
C ++++++ ONEDEE: MODULE 1 ++++++
C ++++++
C *****
C
C PROGRAM ONEDEE EVALUATES BOTH THE FORWARD AND INVERSE PROBLEMS FOR
C THE 1-D (VERTICALLY AVERAGED) FLOW MODEL. THE VERTICAL AVERAGE IS
C OVER THE SECTION OF HOLE CORRESPONDING TO THE BOTTOM OF THE CASING
C TO THE BOTTOM OF THE HOLE. THE HALF-SPACE IS ASSUMED TO GO FROM
C Z=0 TO Z=INFINITY. THE IMPULSE RESPONSE OF THE 1-D AVERAGE IS
C COMPUTED ASSUMING A DELTA-FUNCTION SURFACE PRESSURE SIGNAL. THIS
C IMPULSE RESPONSE IS THEN CONVOLVED WITH THE MEASURED SURFACE
C BAROMETRIC PRESSURE TO OBTAIN THE PREDICTED IN-HOLE PRESSURE. THE
C HALF-SPACE IS ASSUMED TO BE HOMOGENEOUS AND ISOTROPIC.
C THE FLOW MODEL HAS ONLY ONE PARAMETER, NAMELY DIFFUSIVITY. THUS,
C ONLY THE GRID SEARCH OR MONTE CARLO METHODS ARE APPROPRIATE FOR
C THE INVERSE PROBLEM. HOWEVER, IF THERE IS A QUESTION OF EFFECTIVE
C CASING LENGTH (EG. POOR CASING CEMENT JOB), TWO MODEL PARAMETERS
C CAN BE VARIED (CASING LENGTH AND DIFFUSIVITY). AND RIDGE REGRESSION
C CAN ALSO BE USED IN THE INVERSE PROBLEM.
C
C THIS CODE HAS BEEN CAST IN A MORE GENERAL [X,Y(X)] FUNCTIONAL
C NOMENCLATURE. THUS, TIME WILL ALWAYS BE THE X-AXIS AND PRESSURE
C WILL ALWAYS BE THE Y-AXIS.
C
C $NOFLOATCALLS
C $DEBUG
C $LARGE
C
C PROGRAM GFAIP
C "General Forward And Inverse Program"
C
C DIMENSION XINP(2000),XOUT(2000),YINP(2000),YOUT(2000)
C DIMENSION A(10),SIGMA(10),DELTA(10)
C DIMENSION YFIT(2000),SIGMA(2000),B(10),DB(10)
C DIMENSION CHI(20,20),VAR(10,10),COR(10,10)
C DIMENSION XXX(2,2000),YYY(2,2000),X(2000),Y(2000)
C DIMENSION NNN(2),ILINE(2),JLINE(2)
C
C COMMON /ZAP/ JFR(10),NTERMS,NFR
C COMMON /STUFF/XINP,YINP,G(2000),NX
C COMMON /STUFF2/KMIN,KMAX
C
C CHARACTER*20 NTYPE,BZAP,GZAP,NDIS,NPLT,MC,NSEA
C CHARACTER*20 FNAME,RNAME,GNAME,HNAME,NSTOR,ZMEAN
C LOGICAL BFLAG,GFLAG,IFLAG1,IFLAG2
C
C WRITE(*,27000)
C READ(*,35000) NPLT
C
C BFLAG=.FALSE.
C GFLAG=.FALSE.
C
C COMPARE FORWARD MODEL WITH MEASURED OUTPUT DATA
C THIS MUST BE "Y" FOR THE INVERSE PROBLEM. HOWEVER, IT IS
C OPTIONAL FOR THE FORWARD PROBLEM
C WRITE(*,14010)
C READ(*,35000) BZAP
C IF(BZAP.NE.'Y') GOTO 30
C BFLAG=.TRUE.
C
C ENTER MEASURED OUTPUT DATA FILENAME DATA FILENAME
C WRITE(*,19000)

```



```

      READ(*,35000) RNAME
      OPEN(10, FILE=RNAME, STATUS='OLD')
C
C   DOES THE MODEL OUTPUT REQUIRE AN INPUT
C   DATA FILE?
30  WRITE(*,14020)
      READ(*,35000) QZAP
      IF(QZAP.EQ.'N') GOTO 40
      QFLAG=.TRUE.
C
C   ENTER INPUT DATA FILENAME
      WRITE(*,19500)
      READ(*,35000) HNAME
      OPEN(20, FILE=HNAME, STATUS='OLD')
C
40  IF(BZAP.EQ.'N'.AND.QZAP.EQ.'N') GOTO 60
C
C   SUBTRACT MEANS FROM DATA SETS?
      WRITE(*,14500)
      READ(*,35000) ZMEAN
C
      WRITE(*,14000)
      READ(*,*) XSTART,XSTOP
C
      SUM1=0.0
      SUM2=0.0
      MBUMP=0
      WRITE(*,35000)
      READ(*,*) NBUMP
C
C   READ DATA FROM MEASURED INPUT AND/OR OUTPUT DATA FILE
50  IF(BFLAG) READ(10,*) NX2
      IF(QFLAG) READ(20,*) NX1
      X1=0.0
      X2=0.0
      Y1=0.0
      Y2=0.0
      K=0
      NX=MAX0(NX1,NX2)
      DO 60 J=1,NX
        IF(BFLAG) READ(10,*) X2,Y2
        IF(QFLAG) READ(20,*) X1,Y1
        IF(BFLAG.AND.X1.LT.XSTART) GOTO 60
        IF(BFLAG.AND.X1.GT.XSTOP) GOTO 63
        IF(QFLAG.AND.X2.LT.XSTART) GOTO 60
        IF(QFLAG.AND.X2.GT.XSTOP) GOTO 63
        MBUMP=MBUMP+1
        IF(MBUMP.NE.NBUMP) GOTO 60
        K=K+1
        MBUMP=0
C
      INCR=0
      IF(.NOT.BFLAG) GOTO 55
      INCR=1
      XXX(1,K)=X2
      YYY(1,K)=Y2
      XOUT(K)=X2
      YOUT(K)=Y2
      SUM2=SUM2+Y2
55  IF(.NOT.QFLAG) GOTO 60
      XXX(INCR+1,K)=X1
      YYY(INCR+1,K)=Y1
      XINP(K)=X1
      YINP(K)=Y1
      SUM1=SUM1+Y1
60  CONTINUE

```



```

      DX=(XINP(NX)-XINP(1))/FLOAT(NX-1)
C
      DO 200 J=1,NX
        X(J)=XINP(1)+FLOAT(J-1)*DX
      200 CONTINUE
C
      COMPUTE FORWARD MODEL
      250 DO 300 J=1,NX
        Y(J)=FUNCTN(X,J,A)
        XXX(1,J)=X(J)
        YYY(1,J)=Y(J)
        IF(.NOT.BFLAG) GOTO 300
        XXX(2,J)=XOUT(J)
        YYY(2,J)=YOUT(J)
      300 CONTINUE
C
      WRITE(*,14030)
      READ(*,35000) NSTOR
      IF(NSTOR.NE.'Y') GOTO 510
C
      WRITE(*,14040)
      READ(*,35000) GNAME
      OPEN(21,FILE=GNAME,STATUS='NEW')
C
      STORE FORWARD ON FILE
      400 WRITE (21,*) NX
        DO 500 J=1,NX
          WRITE(21,*) X(J),Y(J)
        500 CONTINUE
C
      510 IF (INPLT.NE.'Y') GO TO 2800
C
      ILINE(1)=1
      ILINE(2)=0
      JLINE(1)=7
      JLINE(2)=0
      NNN(1)=NX
      NNN(2)=NX
      IF(BFLAG) NFILES=2
      PAUSE
      CALL XPLOT(NNN,NFILES,ILINE,JLINE,XXX,YYY)
C
      520 GO TO 2800
C
      ----- END OF FORWARD SOLUTION -----
C
      ----- INVERSE SOLUTION -----
C
      ENCODE PARAMETER SET
      INPUT NO. FREE PARAMETERS IN MODEL
      1000 WRITE(*,13000)
        READ(*,37000) NFR
        NFRP1=NFR+1
C
      SPECIFY WHICH PARAMETERS IN MODEL ARE FREE
      DO 1010 J=1,NFR
        WRITE(*,13010) J
        READ(*,37000) JFR(J)
      1010 CONTINUE
C
      ASSIGN REMAINDER OF ENCODED MODEL PARAMETERS
      DO 1020 J=NFRP1,NTERMS
        K=J-NFRP1+1
        WRITE(*,13020) K
        READ(*,37000) JFR(J)
      1020 CONTINUE

```

```

C
C SPECIFY VALUES OF FIXED MODEL PARAMETERS
DO 1030 J=NFRP1,NTERMS
WRITE(*,13040) JFR(J)
READ(*,*) A(J)
1030 CONTINUE
C
C SPECIFY STARTING VALUES OF FREE PARAMETERS
DO 1040 J=1,NFR
WRITE(*,13050) JFR(J)
READ(*,*) A(J)
1040 CONTINUE
C
C SPECIFY ABSCISSA RANGE OVER WHICH ERROR MINIMIZATION
C SHOULD OCCUR
WRITE(*,25005)
READ(*,*) XXMIN,XXMAX
C
C SET UPPER AND LOWER INDICES CORRESPONDING WITH XXMIN, XXMAX
IFLAG1=.FALSE.
IFLAG2=.FALSE.
KMIN=1
KMAX=NPTS
DO 1060 J=1,NPTS
IF(IFLAG1) GOTO 1050
IF(XOUT(J).LT,XXMIN) GOTO 1050
KMIN=J
IFLAG1=.TRUE.
1050 IF(IFLAG2) GOTO 1060
IF(XOUT(J).LT,XXMAX) GOTO 1060
KMAX=J
IFLAG2=.TRUE.
1060 CONTINUE
C
C REDEFINE THE NUMBER OF DEGREES OF FREEDOM
NFREE=(KMAX-KMIN+1)-NFR
C
WRITE(*,25000)
READ(*,35000) MC
C
IF (MC.NE.'Y') GO TO 1500
C
----- BEGIN MONTE CARLO SEARCH -----
C
COUNT=0
TEMP=1.0E20
C INPUT PARAMETER RANGE FOR MONTE CARLO SEARCH
DO 1100 J=1,NFR
WRITE(*,26000) JFR(J)
READ(*,*) B1,B2
B(J)=B1
DB(J)=B2-B1
1100 CONTINUE
C
C INPUT NO. TIMES DICE ARE TO BE THROWN
WRITE(*,10000)
READ(*,*) KMAX
C
C THROW THE DICE
1200 DO 1300 J=1,NFR
A(J)=B(J)+RAN(DUMMY)*DB(J)
1300 CONTINUE
C
DO 1400 J=1,NPTS
YFIT(J)=FUNCTN(X,J,A)
1400 CONTINUE

```

```

C      KOUNT=KOUNT+1
      IF (KOUNT.GT.KMAX) GO TO 2800
      CHISQR=FCHISQ(YOUT,SIGMA,Y,NPTS,NFREE,MODE,YFIT)
      IF (CHISQR.GT.TEMP) GO TO 1200
      PROBAB=PCHISQ(CHISQR,NFREE)
      WRITE (*,41000) CHISQR,(A(K),K=1,NTERMS)
      WRITE (*,41000) PROBAB
      TEMP=CHISQR
      GO TO 1200

C      ----- END OF MONTE CARLO SEARCH -----
C
C      ----- BEGIN NON-LINEAR INVERSION -----
C
C      SET UP PARAMETERS FOR NON-LINEAR INVERSION
1500  NPTS=NX
      FLAMDA=0.001

C
C      SPECIFY FRACTION OF START VALUE TO BE INCREMENT
      WRITE (*,23000)
      READ (*,*) XFRAC

C
C      INITIALIZE PARAMETER INCREMENTS
      DO 1600 J=1,NFR
        DELTAA(J)=XFRAC*A(J)
1600  CONTINUE

C
C      SPECIFY PRECISION OF CHISQR MINIMUM
      WRITE (*,52000)
      READ (*,*) PRES

C
C      SPECIFY EITHER MARQUARDT OR GRID SEARCH
      WRITE (*,22000)
      READ (*,35000) NSEA

C
C      WRITE HEADER
      IF (NSEA.EQ.'M') WRITE (*,16000)
      IF (NSEA.EQ.'G') WRITE (*,17000)

C
C      COMMENCE ITERATION SEQUENCE...STOP WHEN CHISQR IS MINIMUM
      DUMMY=0.0
      TEMP=1.0
      DO 2500 J=1,200
        IF (NSEA.EQ.'G') GOTO 1900
        WRITE (*,42000) J
        CALL CURFIT(XOUT,YOUT,SIGMA,Y,NPTS,NTERMS,NFR,MODE,DELTA,NFREE,VAR,FLAMDA,YFIT,CHISQR)
        X      DELTAA,NFREE,VAR,FLAMDA,YFIT,CHISQR)
        DO 1800 K=1,NFR
          SIGMA(K)=SQRT(VAR(K,X))
          DO 1700 L=1,K
            COR(K,L)=SQRT(VAR(K,L)**2/(VAR(K,X)*VAR(L,L)))
            CONTINUE
1700  CONTINUE
1800  CONTINUE
      GO TO 2000
1900  CALL GRIDLS(XOUT,YOUT,SIGMA,Y,NPTS,NFR,MODE,A,DELTAA,SIGMA,NFREE,YFIT,CHISQR)
        X
2000  PROBAB=PCHISQ(CHISQR,NFREE)

C
      WRITE (*,45000)
      WRITE (*,41010) (JFR(K),K=1,NTERMS)
      WRITE (*,41000) (A(K),K=1,NTERMS)
      WRITE (*,44000) (SIGMA(K),K=1,NTERMS)
      WRITE (*,50000)
      WRITE (*,46000) CHISQR
      WRITE (*,47000) PROBAB

```

```

      IF (NSEA.EQ.'G') GO TO 2400
      WRITE (*,48000)
      DO 2100 K=1,NFR
        WRITE (*,51000) (CDR(K,L),L=1,K)
        CONTINUE
2100   WRITE (*,50000)
        DO 2300 K=2,NFR
          DO 2200 L=1,K-1
            R=CDR(K,L)
            CORREL=1.0- PCORR(R,NPTS,NFR)
            WRITE (*,49000) JFR(K),JFR(L),CORREL
          CONTINUE
2200   CONTINUE
2300   CONTINUE
C
2400   WRITE (*,50000)
C
      RAT=ABS((TEMP-CHISQR)/TEMP)
      IF (RAT.LE.PRES) GO TO 2600
      TEMP=CHISQR
      WRITE (*,43000)
2500   CONTINUE
C
2600   DO 2700 J=1,NPTS
      XXX(1,J)=XOUT(J)
      XXX(2,J)=XOUT(J)
      YYY(1,J)=YOUT(J)
      YYY(2,J)=YFIT(J)
2700   CONTINUE
C
      IF(NPLT.NE.'Y') GOTO 2800
      NFILES=2
      ILINE(1)=1
      ILINE(2)=0
      JLINE(1)=7
      JLINE(2)=0
      NNN(1)=NPTS
      NNN(2)=NPTS
      PAUSE
      CALL XPLOT(NNN,NFILES,ILINE,JLINE,XXX,YYY)
C
2800   STOP
10000  FORMAT(/,' NO. TIMES DICE ARE TO BE THROWN')
12000  FORMAT(/,' FORWARD=F, INVERSE=I')
13000  FORMAT(/,' ENTER NO. FREE PARAMETERS IN MODEL')
13010  FORMAT(/,' ENTER MODEL PARAMETER INDEX CORRESPONDING TO FREE',/,
        '  ' PARAMETER NUMBER',I2)
13020  FORMAT(/,' ENTER MODEL PARAMETER INDEX CORRESPONDING TO FIXED',/,
        '  ' PARAMETER ',I2)
13030  FORMAT(/,' ENTER VALUE OF MODEL PARAMETER ',I2)
13040  FORMAT(/,' ENTER VALUE OF MODEL FIXED PARAMETER ',I2)
13050  FORMAT(/,' ENTER STARTING VALUE OF MODEL FREE PARAMETER ',I2)
14000  FORMAT(/,' ENTER STARTING X, ENDING X')
14010  FORMAT(/,' COMPARE MODEL OUTPUT WITH MEASURED OUTPUT? (Y/N)',/,
        '  ' NOTE: THIS MUST BE "Y" FOR THE INVERSE PROBLEM')
14020  FORMAT(/,' DOES MODEL REQUIRE MEASURED INPUT DATA? (Y/N)')
14030  FORMAT(/,' SAVE MODEL OUTPUT ON A DISK FILE? (Y/N)')
14040  FORMAT(/,' ENTER DATA FILENAME')
14500  FORMAT(/,' SUBTRACT MEAN FROM SIGNALS' (Y/N)')
15000  FORMAT(/,' ENTER NO. DATA POINTS')
16000  FORMAT(1H,10X,' RIDGE REGRESSION ITERATION SEQUENCE',/)
17000  FORMAT(1H,10X,' GRID SEARCH ITERATION SEQUENCE',/)
18000  FORMAT(/,' ENTER PRESSURE DATA FILENAME')
19000  FORMAT(/,' ENTER OUTPUT DATA FILENAME')
19500  FORMAT(/,' ENTER INPUT DATA FILENAME')
22000  FORMAT(/,' MARQUARDT=m, GRID=g')
23000  FORMAT(/,' INPUT FRACTION OF INITIAL R TO BE DELTA-R')

```

```

24000 FORMAT(1H,2(E12.4,2X))
25000 FORMAT(/,' MONTE CARLO SEARCH? (Y/N)')
25005 FORMAT(/,' SPECIFY XMIN,XMAX OVER WHICH ERROR MINIMIZATION ',/,
X      ' SHOULD OCCUR')
26000 FORMAT(/,' INPUT MIN,MAX VALUES FOR MODEL PARAMETER ',12)
27000 FORMAT(/,' DO PLOTS? (Y/N)')
27500 FORMAT(/,' ENTER TOTAL NUMBER OF MODEL PARAMETERS')
29000 FORMAT(/,' NOISY DATA? (Y/N)')
30000 FORMAT(/,' INPUT NOISE LEVEL')
34000 FORMAT(1H,'A(',11,')=' ,E12.4)
35000 FORMAT(A)
35000 FORMAT(/,' ENTER DATA SPACING (EG. EVERY POINT = 1. EVERY ',/,
X      ' OTHER POINT = 2. ETC.')
```

```

37000 FORMAT(12)
39000 FORMAT(14)
41000 FORMAT(1H,3(E11.4,10X))
41010 FORMAT(1H,3(' MODEL PARAM.',12,5X),/)
42000 FORMAT(/,' ***** ITERATION ',13,' *****',/)
43000 FORMAT(/,' *****',/)
44000 FORMAT(1H,3(' ',E11.4,')',8X),/)
45000 FORMAT(/,' PARAMETERS AND UNCERTAINTIES',/)
46000 FORMAT(/,' REDUCED CHI-SQUARE = ',E12.4)
47000 FORMAT(/,' PROBABILITY OF CORRECT PARENT FUNCTION = ',E12.4,/)
48000 FORMAT(1H,10X,'FREE PARAMETER CORRELATION MATRIX',/)
49000 FORMAT(1H,' PROBABILITY THAT A(' ,11,') IS CORRELATED WITH
X A(' ,11,') IS ',E12.4,/)
50000 FORMAT(1H )
51000 FORMAT(1H,5(F12.4,2X))
52000 FORMAT(/,' ENTER PRECISION')
55000 FORMAT(1H,12,2X,12,5X,4(E12.4,5X))
56000 FORMAT(1H,3(E12.4,2X))
END
FUNCTION PCORRE(R,NPTS,NTERMS)

C
C   PURPOSE
C       EVALUATE PROBABILITY FOR NO CORRELATION BETWEEN TWO VARIABLES
C
C   DESCRIPTION OF PARAMETERS
C       R      - LINEAR CORRELATION COEFFICIENT
C       NPTS   - NUMBER OF DATA POINTS
C       NTERMS - NUMBER OF MODEL PARAMETERS
C
C   SUBROUTINES AND FUNCTION SUBPROGRAMS REQUIRED
C       GAMMA(X)
C           CALCULATES GAMMA FUNCTION FOR INTEGERS AND HALF-INTEGERS
C
C       -----
C       COMMON/STUFF2/KMIN,KMAX
C       -----
C       DOUBLE PRECISION R2,TERM,SUM,FI,FNUM,DENOM
C
C       EVALUATE NUMBER OF DEGREES OF FREEDOM
100  NFREE=(KMAX-KMIN+1)-NTERMS
    IF (NFREE) 200,200,300
200  PCORRE=0.0
    GO TO 1800
300  R2=R**2
    IF (1.0-R2) 200,200,400
400  NEVEN=2*(NFREE/2)
    IF (NFREE-NEVEN) 500,500,1000
C
C   NUMBER OF DEGREES OF FREEDOM IS EVEN
500  IMAX=(NFREE-2)/2
    FREE=NFREE
600  TERM=ABS(R)
    SUM=TERM
```

```

      IF (IMAX) 1600,700,800
700 PCORRE=1.0-TERM
      GO TO 1800
800     DO 900 I=1,IMAX
          FI=I
          FNUM=IMAX-I+1
          DENOM=2*I+1
          TERM=-TERM+R2*FNUM/FI
900     SUM=SUM+TERM/DENOM
      PCORRE=1.128379167*(GAMMA((FREE+1.0)/2.0)/GAMMA(FREE/2.0))
      PCORRE=1.0-PCORRE*SUM
      GO TO 1600

      NUMBER OF DEGREES OF FREEDOM IS DDD

1000 IMAX=(NFREE-3)/2
1100 TERM=ABS(R)*DSQRT(1.0-R2)
1200 SUM=DATAN(R2/TERM)
      IF (IMAX) 1700,1300,1400
1300 SUM=SUM+TERM
      GO TO 1700
1400 SUM=SUM+TERM
1500     DO 1600 I=1,IMAX
          FNUM=2*I
          DENOM=2*I+1
          TERM=TERM*(1.0-R2)*FNUM/DENOM
1600     SUM=SUM+TERM
1700 PCORRE=1.0-0.6366197724*SUM
1800 RETURN
      END
      FUNCTION PCHISQ (CHISQ,NFREE)

C
C      PURPOSE
C          EVALUATE PROBABILITY FOR EXCEEDING CHI SQUARE
C
C      USAGE
C          RESULT = PCHISQ (CHISQ, NFREE)
C
C      DESCRIPTION OF PARAMETERS
C          CHISQ - COMPARISON VALUE OF REDUCED CHI SQUARE
C          NFREE - NUMBER OF DEGREES OF FREEDOM
C
C      SUBROUTINES AND FUNCTION SUBPROGRAMS REQUIRED
C          GAMMA(X)
C          CALCULATES GAMMA FUNCTION FOR INTEGERS AND HALF-
C          INTEGERS
C
C      COMMENTS
C          CALCULATION IS APPROXIMATE FOR NFREE ODD AND CHI SQUARE
C          GREATER THAN 50
C
      DOUBLE PRECISION Z,TERM,SUM,FACT
100 IF (NFREE) 200,200,300
200 PCHISQ=0.
      GO TO 1500
300 FREE=NFREE
      Z=CHISQ*FREE/2.
      NEVEN=2*(NFREE/2)
      IF (NFREE-NEVEN) 400,400,800

C
C          NUMBER OF DEGREES OF FREEDOM IS EVEN
C
400 IMAX=NFREE/2
      TERM=1.
      SUM=0.
500     DO 600 I=1,IMAX

```



```

        FI=I
        SUM=SUM+TERM
        IF (ABS(SUM).GT.1.0E15) GO TO 650
600    TERM=TERM*Z/FI
650    FACT=0.000
        IF (DABS(Z).GT.15.000) GO TO 700
        FACT=DEXP(-Z)
700    PCHISQ=SUM*FACT
        GO TO 1500

C
C      NUMBER OF DEGREES OF FREEDOM IS GDD
C
800    IF (Z-25) 1000,1000,900
900    Z=CHISQ*(FREE-1.)/2.
        GO TO 400
1000   PWR=FREE/2.
        TERM=1.
        SUM=TERM/PWR
1100    DO 1300 I=1,1000
        FI=I
        TERM=-TERM*Z/FI
        SUM=SUM+TERM/(PWR+FI)
1200    IF (DABS(TERM/SUM)-0.00001) 1400,1400,1300
1300    CONTINUE
1400    PCHISQ=1.-(Z**PWR)*SUM/GAMMA(PWR)
1500    RETURN
        END
        FUNCTION GAMMA(X)

C
C      PURPOSE
C      CALCULATE THE GAMMA FUNCTION FOR INTEGERS AND HALF-
C      INTEGERS
C
C      USAGE
C      RESULT=GAMMA(X)
C
C      DESCRIPTION OF PARAMETERS
C      X - INTEGER OR HALF-INTEGERS
C
C      SUBROUTINES AND FUNCTION SUBPROGRAMS REQUIRED
C      FACTOR(N)
C      CALCULATES N FACTORIAL FOR INTEGERS
C
C      DOUBLE PRECISION PROD,SUM,FI
C
C      INTEGERIZE ARGUMENT
C
100    N=X-0.25
        XN=N
200    IF (X-XN-0.75) 400,400,300
C
C      ARGUMENT IS INTEGER
C
300    GAMMA=FACTOR(N)
        GO TO 1200
C
C      ARGUMENT IS HALF-INTEGERS
C
400    PROD=1.77245385
        IF (N) 800,800,500
500    IF (N-10) 600,600,900
600    DO 700 I=1,N
        FI=I
700    PROD=PROD*(FI-0.5)
800    GAMMA=PROD
        GO TO 1200

```

```

500 SUM=0.0
      DO 1000 I=11,N
        FI=I
1000    SUM=SUM+DLOG(FI-0.5)
1100    GAMMA=PROD*639383.8623*DEXP(SUM)
1200    RETURN
      END
      FUNCTION FACTOR(N)
C
C      PURPOSE
C      CALCULATES FACTORIAL FUNCTION FOR INTEGERS
C
C      DESCRIPTION OF PARAMETERS
C      N - INTEGER ARGUMENT
C
C      DOUBLE PRECISION FI,SUM
100    FACTOR=1.
      IF (N-1) 600,800,200
200    IF (N-10) 300,300,500
C
C      N LESS THAN 11
C
300      DO 400 I=2,N
        FI=I
400      FACTOR=FACTOR*FI
      GO TO 800
C
C      N GREATER THAN 10
C
500    SUM=0.0
      DO 600 I=11,N
        FI=I
600      SUM=SUM+DLOG(FI)
700    FACTOR=3628800.*DEXP(SUM)
800    RETURN
      END
      SUBROUTINE MATINV(ARRAY,NORDER,DET)
C
C      PURPOSE
C      INVERT A SYMMETRIC MATRIX AND CALCULATE ITS DETERMINANT
C
C      DESCRIPTION OF PARAMETERS
C      ARRAY - INPUT MATRIX WHICH IS REPLACED BY ITS INVERSE
C      NORDER - DEGREE OF MATRIX (ORDER OF DETERMINANT)
C      DET - DETERMINANT OF INPUT MATRIX
C
C      COMMENTS
C      DIMENSION STATEMENT VALID FOR NORDER UP TO 10
C
      DOUBLE PRECISION ARRAY,AMAX,SAVE
      DIMENSION ARRAY(10,10),IK(10),JK(10)
100    DET=1.
200      DO 2500 K=1,NORDER
C
C      FIND LARGEST ELEMENT ARRAY(I,J) IN REST OF MATRIX
C
      AMAX=0.
300      DO 600 I=K,NORDER
        DO 600 J=K,NORDER
          IF (DABS(AMAX)-DABS(ARRAY(I,J))) 500,500,600
500      AMAX=ARRAY(I,J)
          IK(K)=I
          JK(K)=J
600      CONTINUE
C
C      INTERCHANGE ROWS AND COLUMNS TO PUT AMAX IN ARRAY(K,K)

```

```

C
700     IF (AMAX) 900,800,900
800     DET=0.
      GO TO 3300
900     I=IK(K)
      IF (I-K) 300,1200,1000
1000      DO 1100 J=1,NORDER
          SAVE=ARRAY(K,J)
          ARRAY(K,J)=ARRAY(I,J)
1100      ARRAY(I,J)=-SAVE
1200      J=JK(K)
      IF (J-K) 300,1500,1300
1300      DO 1400 I=1,NORDER
          SAVE=ARRAY(I,K)
          ARRAY(I,K)=ARRAY(I,J)
1400      ARRAY(I,J)=-SAVE
C
C     ACCUMULATE ELEMENTS OF INVERSE MATRIX
C
1500      DO 1700 I=1,NORDER
      IF (I-K) 1600,1700,1600
1600      ARRAY(I,K)=-ARRAY(I,K)/AMAX
1700      CONTINUE
1800      DO 2100 I=1,NORDER
      DO 2100 J=1,NORDER
      IF (I-K) 1900,2100,1900
      IF (J-K) 2000,2100,2000
2000      ARRAY(I,J)=ARRAY(I,J)+ARRAY(I,K)*ARRAY(K,J)
2100      CONTINUE
2200      DO 2400 J=1,NORDER
      IF (J-K) 2300,2400,2300
2300      ARRAY(K,J)=ARRAY(K,J)/AMAX
2400      CONTINUE
      ARRAY(K,K)=1./AMAX
2500      DET=DET*AMAX
C
C     RESTORE ORDERING OF MATRIX
C
2600      DO 3200 L=1,NORDER
      K=NORDER-L+1
      J=IK(K)
      IF (J-K) 2900,2900,2700
2700      DO 2800 I=1,NORDER
          SAVE=ARRAY(I,K)
          ARRAY(I,K)=-ARRAY(I,J)
2800      ARRAY(I,J)=SAVE
2900      I=JK(K)
      IF (I-K) 3200,3200,3000
3000      DO 3100 J=1,NORDER
          SAVE=ARRAY(K,J)
          ARRAY(K,J)=-ARRAY(I,J)
3100      ARRAY(I,J)=SAVE
3200      CONTINUE
3300      RETURN
      END
      SUBROUTINE CURFIT(X,Y,SIGMAY,NPTS,NTOT,NTERMS,MODE,A,DELTA)
      X NFREE,VAR,FLAMDA,YFIT,CHISQR)
      DOUBLE PRECISION ARRAY
      DIMENSION X(1),Y(1),SIGMAY(1),A(1),DELTA(1),
      X YFIT(1)
      DIMENSION WEIGHT(1000),ALPHA(10,10),BETA(10),DERIV(10),
      X ARRAY(10,10),S(10),VAR(10,10)
C
C     PURPOSE:
C     MAKE A LEAST-SQUARES FIT TO A NON-LINEAR FUNCTION
C

```

```

C      USAGE:
C      CALL CURFIT(X,Y,SIGMAY,NPTS,NTERMS,NTOT,MODE,A,DELTA,
C      X SIGMA,FLAMDA,YFIT,CHISQR)
C
C      DESCRIPTION OF PARAMETERS
C      X      - ARRAY OF DATA POINTS FOR INDEPENDENT VARIABLE
C      Y      - ARRAY OF DATA POINTS FOR DEPENDENT VARIABLE
C      SIGMAY - ARRAY OF STANDARD DEVIATIONS FOR Y DATA POINTS
C      NPTS   - NUMBER OF PAIRS OF DATA POINTS
C      NTERMS - NUMBER OF FREE PARAMETERS
C      NTOT   - TOTAL NUMBER OF MODEL PARAMETERS
C      MODE   - DETERMINES METHOD OF WEIGHTING LEAST-SQUARES FIT
C              +1 (INSTRUMENTAL) WEIGHT(I)=1./SIGMAY(I)**2
C              0 (NO WEIGHTING) WEIGHT(I)=1.
C              -1 (STATISTICAL) WEIGHT(I)=1./Y(I)
C      A      - ARRAY OF PARAMETERS
C      DELTA  - ARRAY OF INCREMENTS FOR PARAMETERS A
C      VAR    - VARIANCE-COVARIANCE MATRIX FOR PARAMETERS A
C      FLAMDA - PROPORTION OF GRADIENT SEARCH INCLUDED
C      YFIT   - ARRAY OF CALCULATED VALUES OF Y
C      CHISQR - REDUCED CHI SQUARE FOR FIT
C
C      SUBROUTINES AND FUNCTION SUBPROGRAMS REQUIRED
C      FUNCTN(X,I,A)
C          EVALUATES THE FITTING FUNCTION FOR THE ITH TERM
C      FCHISQ(Y,SIGMAY,NPTS,NFREE,MODE,YFIT)
C          EVALUATES REDUCED CHI SQUARE FOR FIT TO DATA
C      FDERIV(X,I,A,DELTA,NTERMS,DERIV)
C          EVALUATES THE DERIVATIVES OF THE FITTING FUNCTION
C          FOR THE ITH TERM WITH RESPECT TO EACH PARAMETER
C      MATINV(ARRAY,NTERMS,DET)
C          INVERTS A SYMMETRIC TWO-DIMENSIONAL MATRIX OF DEGREE NTERMS
C          AND CALCULATES ITS DETERMINANT
C
C      COMMENTS:
C          DIMENSION STATEMENT VALID FOR NTERMS UP TO 10
C          SET FLAMDA=0.001 AT BEGINNING OF SEARCH
C
C      INITIALIZE B MATRIX
C      DO 50 J=1,NTOT
C      B(J)=A(J)
C      50 CONTINUE
C
C      100 NFREE= NPTS-NTERMS
C      IF (NFREE) 200,200,300
C      200 CHISQR=0.
C      GO TO 3400
C
C      EVALUATE WEIGHTS
C
C      300 DO 1000 I=1,NPTS
C      400 IF (MODE) 500,800,900
C      500 IF (Y(I)) 700,800,600
C      600 WEIGHT(I)=1./Y(I)
C      GO TO 1000
C      700 WEIGHT(I)=1./(-Y(I))
C      GO TO 1000
C      800 WEIGHT(I)=1.
C      GO TO 1000
C      900 WEIGHT(I)=1./SIGMAY(I)**2
C      1000 CONTINUE
C
C      EVALUATE ALPHA AND BETA MATRICES
C
C      1100 DO 1200 J=1,NTERMS
C      BETA(J)=0.

```

```

      DO 1200 K=1,J
1200      ALPHA(J,K)=0.
1300      DO 1500 I=1,NPTS
      CALL FDERIV(X,I,A,DELTAA,NTERMS,DERIV)
      DO 1400 J=1,NTERMS
      BETA(J)=BETA(J)+WEIGHT(I)*(Y(I)-FUNCTN(X,I,A))*DERIV(J)
      DO 1400 K=1,J
      ALPHA(J,K)=ALPHA(J,K)+WEIGHT(I)*DERIV(J)*DERIV(K)
1400
1500      CONTINUE
1600      DO 1700 J=1,NTERMS
      DO 1700 K=1,J
1700      ALPHA(K,J)=ALPHA(J,K)
C
C      EVALUATE CHISQUARE AT STARTING POINT
C
1800      DO 1900 I=1,NPTS
1900      YFIT(I)=FUNCTN(X,I,A)
2000      CHISQ1=FCHISQ(Y,SIGMAY,NPTS,NFREE,MODE,YFIT)
C
C      INVERT MODIFIED CURVATURE MATRIX TO FIND NEW PARAMETERS
C
2100      DO 2300 J=1,NTERMS
      DO 2200 K=1,NTERMS
      ARRAY(J,K)=ALPHA(J,K)/SQRT(ALPHA(J,J)*ALPHA(K,K))
2200      ARRAY(J,J)=1.+FLAMDA
2300      CALL MATINV(ARRAY,NTERMS,DET)
2400      DO 2600 J=1,NTERMS
      B(J)=A(J)
      DO 2600 K=1,NTERMS
      VAR(J,K)=ARRAY(J,K)/(SQRT(ALPHA(J,J)*ALPHA(K,K)))
      B(J)=B(J)+BETA(K)*VAR(J,K)
2600
C
C      IF CHI SQUARE INCREASED, INCREASE FLAMDA AND TRY AGAIN
C
2700      DO 2800 I=1,NPTS
2800      YFIT(I)=FUNCTN(X,I,B)
2900      CHISQ2=FCHISQ(Y,SIGMAY,NPTS,NFREE,MODE,YFIT)
      IF (CHISQ1-CHISQ2) 3000,3100,3100
3000      FLAMDA=10.0*FLAMDA
      GO TO 2100
C
C      EVALUATE PARAMETERS AND VARIANCE-COVARIANCE MATRIX
C
3100      DO 3300 J=1,NTERMS
      DO 3200 K=1,NTERMS
      VAR(J,K)=CHISQ2*VAR(J,K)
3200      A(J)=B(J)
3300      FLAMDA=FLAMDA/10.0
3400      RETURN
      END
      SUBROUTINE FDERIV(X,I,A,DELTAA,NTERMS,DERIV)
      DIMENSION X(1),A(1),DELTAA(1),DERIV(1)
C
C      PURPOSE:
C      EVALUATE DERIVATIVES OF FUNCTION FOR LEAST-SQUARES
C      SEARCH FOR ARBITRARY FUNCTION GIVEN BY FUNCTN
C
C      USAGE:
C      CALL FDERIV(X,I,A,DELTAA,NTERMS,DERIV)
C
C      DESCRIPTION OF PARAMETERS
C      X      - ARRAY OF DATA POINTS FOR INDEPENDENT VARIABLE
C      I      - INDEX OF DATA POINTS
C      A      - ARRAY OF PARAMETERS
C      DELTAA - ARRAY OF PARAMETER INCREMENTS
C      NTERMS - NUMBER OF PARAMETERS

```

```

C      DERIV      - DERIVATIVES OF FUNCTION
C
C      SUBROUTINE AND FUNCTION SUBPROGRAMS REQUIRED
C      FUNCTN(X,I,A)
C      EVALUATES THE FITTING FUNCTION FOR THE ITH TERM
C
100      DO 200 J=1,NTERMS
          AJ=A(J)
          DELTA=DELTA(J)
          A(J)=AJ+DELTA
          YFIT=FUNCTN(X,I,A)
          A(J)=AJ-DELTA
          DERIV(J)=(YFIT-FUNCTN(X,I,A))/(2.*DELTA)
200      A(J)=AJ
          RETURN
          END
          FUNCTION FCHISQ(Y,SIGMAY,NPTS,NFREE,MODE,YFIT)
          DOUBLE PRECISION CHISQ,WEIGHT
          DIMENSION Y(1),SIGMAY(1),YFIT(1)
C
C      -----
C      COMMON /STUFF2/KMIN,KMAX
C      -----
C
C      PURPOSE
C      EVALUATE REDUCED CHI SQUARE FOR FIT TO DATA
C      FCHISQ=SUM((Y-YFIT)**2/SIGMA**2)/NFREE
C
C      USAGE
C      RESULT=FCHISQ(Y,SIGMAY,NPTS,NFREE,MODE,YFIT)
C
C      DESCRIPTION OF PARAMETERS
C      Y      - ARRAY OF DATA POINTS
C      SIGMAY  - ARRAY OF STANDARD DEVIATIONS FOR DATA POINTS
C      NPTS    - NUMBER OF DATA POINTS
C      NFREE   - NUMBER OF DEGREES OF FREEDOM
C      MODE    - DETERMINES METHOD OF WEIGHTING LEAST-SQUARES FIT
C                - +1 (INSTRUMENTAL) WEIGHT(I)=1./SIGMAY(I)**2
C                - 0 (NO WEIGHTING) WEIGHT(I)=1.
C                - -1 (STATISTICAL) WEIGHT(I)=1./Y(I)
C
C      YFIT   - ARRAY OF CALCULATED VALUES OF Y
C
100      CHISQ=0.
200      IF (NFREE) 300,300,400
300      FCHISQ=0.
          GO TO 1400
C
C      ACCUMULATE CHI SQUARE
C
C      -----
400      DO 1100 I=KMIN,KMAX
C      -----
500      IF (MODE) 600,900,1000
600      IF (Y(I)) 600,900,700
700      WEIGHT=1./Y(I)
          GO TO 1100
800      WEIGHT=1./(-Y(I))
          GO TO 1100
900      WEIGHT=1.
          GO TO 1100
1000     WEIGHT=1./SIGMAY(I)**2
          CHISQ=CHISQ+WEIGHT*(Y(I)-YFIT(I))**2
1100     CONTINUE
C
C      DIVIDE BY NUMBER OF DEGREES OF FREEDOM

```

```

C
1200 FREE=NFREE
1300 FCHISQ=CHISQ/FREE
      WRITE(*,2000) FCHISQ
2000 FORMAT(/,' REDUCED CHI-SQUARE IS ',E12.4)
1400 RETURN
      END
C      SUBROUTINE GRIDLS
C
C      PURPOSE
C      MAKE A GRID-SEARCH LEAST-SQUARES FIT TO DATA WITH A SPECIFIED
C      FUNCTION WHICH IS NOT LINEAR IN COEFFICIENTS
C
C      USAGE
C      CALL GRIDLS(X,Y,SIGMA,NPTS,NTERMS,MODE,A,DELTA,SIGMA,
C      NFREE,YFIT,CHISQ)
C
C      DESCRIPTION OF PARAMETERS
C      X      - ARRAY OF DATA POINTS FOR INDEPENDENT VARIABLE
C      Y      - ARRAY OF DATA POINTS FOR DEPENDENT VARIABLE
C      SIGMA   - ARRAY OF STANDARD DEVIATIONS FOR Y DATA POINTS
C      NPTS    - NUMBER OF PAIRS OF DATA POINTS
C      NTERMS  - NUMBER OF PARAMETERS
C      MODE    - DETERMINES METHOD OF WEIGHTING LEAST-SQUARES FIT
C      +1 (INSTRUMENTAL) WEIGHT(I)=1.0/SIGMA(I)**2
C      0 (NO WEIGHTING) WEIGHT(I)=1.0
C      -1 (STATISTICAL) WEIGHT(I)=1.0/Y(I)
C      A      - ARRAY OF PARAMETERS
C      DELTA   - ARRAY OF INCREMENTS FOR PARAMETERS A
C      SIGMA   - ARRAY OF STANDARD DEVIATIONS FOR PARAMETERS A
C      YFIT    - ARRAY OF CALCULATED VALUES OF Y
C      CHISQ   - REDUCED CHI SQUARE FOR FIT
C
C      SUBROUTINES AND FUNCTION SUBPROGRAMS REQUIRED
C      FUNCTN(X,I,A)
C      EVALUATES THE FITTING FUNCTION FOR THE ITH TERM
C      FCHISQ(Y,SIGMA,NPTS,NFREE,MODE,YFIT)
C      EVALUATES REDUCED CHI SQUARE FOR FIT TO DATA
C
C      COMMENTS
C      DELTA VALUES ARE MODIFIED BY THE PROGRAM
C
C      SUBROUTINE GRIDLS (X,Y,SIGMA,NPTS,NTERMS,MODE,A,DELTA,
C      SIGMA,NFREE,YFIT,CHISQ)
C
C      DIMENSION X(1),Y(1),SIGMA(1),A(1),DELTA(1),SIGMA(1),YFIT(1)
C      COMMON/STUFF2/KMIN,KMAX
C
C      100 FREE=NFREE
C      CHISQ=0.0
C      IF (NFREE) 2500,2500,2000
C      200 DO 2100 J=1,NTERMS
C
C      EVALUATE CHI SQUARE AT FIRST TWO SEARCH POINTS
C
C
C      300 DO 400 I=1,NPTS
C      400 YFIT(I)=FUNCTN(X,I,A)
C      500 CHISQ1=FCHISQ(Y,SIGMA,NPTS,NFREE,MODE,YFIT)
C      FN=0.0
C      DELTA=DELTA(J)
C      600 A(J)=A(J)+DELTA
C      DO 700 I=1,NPTS
C      700 YFIT(I)=FUNCTN(X,I,A)
C      800 CHISQ2=FCHISQ(Y,SIGMA,NPTS,NFREE,MODE,YFIT)
C      900 IF (CHISQ1-CHISQ2) 1000,600,1300

```

```

C
C      REVERSE DIRECTION OF SEARCH IF CHI SQUARE IS INCREASING
C
1000      DELTA=-DELTA
          A(J)=A(J)+DELTA
              DO 1100 I=1,NPTS
1100          YFIT(I)=FUNCTN(X,I,A)
          SAVE=CHISQ1
          CHISQ1=CHISQ2
1200          CHISQ2=SAVE
C
C      INCREMENT A(J) UNTIL CHI SQUARE INCREASES
C
1300      FN=FN+1.0
          A(J)=A(J)+DELTA
              DO 1400 I=1,NPTS
1400          YFIT(I)=FUNCTN(X,I,A)
          CHISQ3=FCHISQ(Y,SIGMAY,NPTS,NFREE,MODE,YFIT)
1500          IF (CHISQ3-CHISQ2) 1600,1700,1700
1600          CHISQ1=CHISQ2
          CHISQ2=CHISQ3
          GO TO 1300
C
C      FIND MINIMUM OF PARABOLA DEFINED BY LAST THREE POINTS
C
1700      DELTA=DELTA*(1./(1.+(CHISQ1-CHISQ2)/(CHISQ3-CHISQ2)))+0.5)
1800      A(J)=A(J)-DELTA
1900      SIGMAA(J)=DELTAA(J)*SQRT 2.0/(FREE*(CHISQ3-2.0*CHISQ2+CHISQ1)
      X
          )
2000      DELTAA(J)=DELTAA(J)*FN/3.0
2100      CONTINUE
C
C      EVALUATE FIT AND CHI SQUARE FOR FINAL PARAMETERS
C
2200      DO 2300 I=1,NPTS
2300          YFIT(I)=FUNCTN(X,I,A)
2400      CHISQR=FCHISQ(Y,SIGMAY,NPTS,NFREE,MODE,YFIT)
          DO 2450 K=1,NTERMS
2450          SIGMAA(K)=SQRT(CHISQR)*SIGMAA(K)
2500      RETURN
      END
      FUNCTION ERF(X)
      ERF=1.0
      IF (X.GT.10) GO TO 10
      P=0.3275911
      A1=0.254829592
      A2=-0.284495736
      A3=1.421413741
      A4=-1.453152027
      A5=1.061405429
      T=1.0/(1.0+P*X)
      F1=A1+T*(A2+T*(A3+T*(A4+T*A5)))
      F1=T*F1
      ERF=1.0-F1*EXP(-X*X)
10      RETURN
      END

```



```

20 CONTINUE
C
C CONVOLVE IMPULSE RESPONSE WITH SYSTEM INPUT
25 SUM=0.0
YINP(NPTS+1)=YINP(NPTS)
DO 30 J=1,I
SUM=SUM+G(J)*(YINP(I-J+1)+YINP(I-J+2))/2.0
30 CONTINUE
FUNCTN=SUM
RETURN
END

```