isvr Institute of Sound and Vibration Research

TEA – A Suite of Computer Programs for Elastodynamic Analysis Using Coupled Boundary and Finite Elements

C.J.C. Jones, D.J. Thompson and M. Petyt

ISVR Technical Memorandum 840

August 1999

University of Southampton

SCIENTIFIC PUBLICATIONS BY THE ISVR

Technical Reports are published to promote timely dissemination of research results by ISVR personnel. This medium permits more detailed presentation than is usually acceptable for scientific journals. Responsibility for both the content and any opinions expressed rests entirely with the author(s).

Technical Memoranda are produced to enable the early or preliminary release of information by ISVR personnel where such release is deemed to the appropriate. Information contained in these memoranda may be incomplete, or form part of a continuing programme; this should be borne in mind when using or quoting from these documents.

Contract Reports are produced to record the results of scientific work carried out for sponsors, under contract. The ISVR treats these reports as confidential to sponsors and does not make them available for general circulation. Individual sponsors may, however, authorize subsequent release of the material.

UNIVERSITY OF SOUTHAMPTON

INSTITUTE OF SOUND AND VIBRATION RESEARCH

DYNAMICS GROUP

TEA – A Suite of Computer Programs for Elastodynamic Analysis Using Coupled Boundary and Finite Elements

by

C.J.C. Jones, D.J. Thompson and M. Petyt

ISVR Technical Memorandum No. 840

August 1999

Authorized for issue by
Dr. M.J. Brennan
Group Chairman

SUMMARY

A coupled Finite Element (FE) and Boundary Element (BE) model for vibration propagation in two-dimensional solids (plane strain) has been produced. The primary use of the software is the modelling of propagation of ground vibration due to railways.

A coupled BE/FE model may comprise a number of domains each of which may be modelled either using finite elements or boundary elements. For the boundary element domains special provision for open boundaries has been made so that the infinite ground surface or infinite interfaces between ground layers can be modelled as open boundaries of finite length. This approach has already been established in earlier work on single boundary element domains. The theory implemented in the computer program is described in a separate report. Triangle and quadrilateral elements with both linear and quadratic shape functions have been implemented for the finite element parts of the model. A single three-noded boundary element with quadratic shape functions is implemented.

This report describes the software package and provides instruction for its use. A number of example analyses are presented for the special cases of a homogeneous half-space and a layer overlying a stiffer half-space. A coupled FE/BE model is examined in each case and results are compared with a semi-analytical model so that the software is validated and the numerical error associated with the modelling method is investigated. It is found that, with the elements provided, accurate solutions to realistic problems may be obtained for the vibration response up to 200 Hz using a PC with 128 Mb of memory.

CONTENTS

1. INTRODUCTION **1**

2. PROGRAM STRUCTURE **1**

3. DATA PREPARATION **4**

3.1 GENERAL METHOD OF DATA SPECIFICATION 5
3.2 THE **TITLE* MODULE 6
3.3 THE **MATERIAL* MODULE 6
3.4 THE **FREQUENCY* MODULE 6
3.5 THE **NODES* MODULE 6
3.6 THE **ANSYS* MODULE HEADER 6
3.7 THE **FINITE ELEMENTS* MODULE 7
3.8 THE **BOUNDARY ELEMENTS* MODULE 8
3.9 THE **LOADS* MODULE 9
3.10 THE **END* HEADER 9
3.11 THE ** HEADER 9
3.12 CHECKING OF INPUT DATA AND PLOTTING THE STRUCTURE. 10
3.13 RESULTS FILE 10

4. EXAMPLE RESULTS **10**

4.1 PARAMETERS FOR THE EXAMPLES 10
4.2 HOMOGENEOUS HALF-SPACE 11
4.3 LAYERED HALF-SPACE 19

5. CONCLUSIONS **23**

6. ACKNOWLEDGEMENT **23**

7. REFERENCES **24**

Appendix A Example data file.

Appendix B Include files that dimension array space in *teapot, teacake* and *teabags*.

1. INTRODUCTION

It is essential in the analysis of ground vibration using numerical models to ensure that the method used properly represents travelling wave behaviour, *i.e.* the radiation of vibration waves in an infinite visco-elastic medium. The boundary element method is ideally suited to this purpose [1] whilst also providing the ability to model arbitrary geometry of the ground surface and interfaces. Previously a frequency domain model has been produced that demonstrates the method capable of modelling vibration propagation for the structure-borne noise ('ground-borne noise') frequency range up to 200 Hz [2] when quadratic boundary elements are used. The model of reference [2], however, was limited to a single homogeneous medium. For 'thin' but finite structures such as tunnel walls, etc. the boundary element is not efficient as very small elements must be used. In these parts of the model the finite element method is well suited to the purpose. This report describes the software 'TEA' (Two-dimensional Elastodynamic Analysis) which implements a coupled finite element/boundary element model in two-dimensions (plane strain) with multiple boundary element domains.

The present report provides instructions for the use of the software. The input data file is described and some results are examined in comparison with those from a semi-analytical model for a two-dimensional half-space or layered half-space model of the ground [4]. By this means the nature of the error in the results and method of using the program to minimise the error is examined. The theory implemented in the software is described in a separate report [3]. The boundary element (BE) domain modelling is based on the method described in [2, 3, 5].

2. PROGRAM STRUCTURE

The program, TEA, consists of a number of component executables which are run under MSDOS on a standard PC. The software, by its nature, has a large computer memory requirement. In order to minimise this and to allow investigation of problems in the preparation of the data representing the model, the process of setting up the matrices for the subdomains of the problem and assembling them into a global matrix, the software comprises three main programs that are run in sequence by a batch file named *tea.bat*. An annotated listing of this file is given below (see also Figure 1).

The separate component executables run by the batch file and the data files they read and produce are described in Table 1.

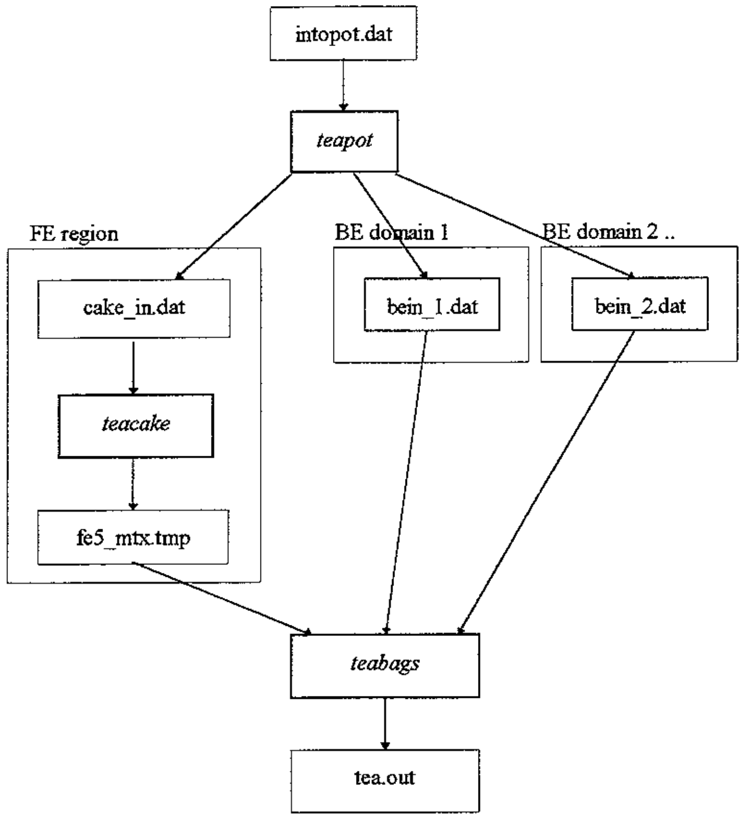


Figure 1. Scheme of programs implementing the model (program names in italics).

rem this program from MSDOS prompt using rem the following command rem tea [input path/file] [output path/file]	These appear as %1 and %2 below.
del *.tmp del *.dat del *.plt del *.out	Delete the files created by previous runs.
copy %1 intopot.dat	Copy the input data file into this directory.
teapot	Read data and create temporary files for creation of FE matrices and BE matrices.
if not exist fetop.plt goto BAGS	This file is not created if there are no finite elements specified in the model. Therefore do not run the program to calculate finite element matrices.
teacake <cake_in.dat >cake.out	Calculate FE matrices.
:BAGS	Jump to here to miss FE matrix calculation.
teabags	Run the global assembly and solution program.
copy tea.out %2	Copy main output file to user specified destination.

Table 1. Description of the separate executables and the files used for passing data between them.

TEAPOT.exe	<u>**P**arse, **O**rder and **T**ranscribe data</u> This executable reads the main data file and performs some checks for consistency. A compact 'local' node numbering scheme is derived for each sub-domain of the model and a list of the local numbering versus the global numbering scheme is compiled. In the case of the finite elements, the local node numbering is ordered so that node numbers increase from the lower edge of the model upwards and from the right-hand edge leftwards. The purpose of this is to produce finite element matrices with a limited bandwidth. The executable then writes the data to a number of separate data files for the use of the other programs. Files read: intopot.dat — This is a copy of the input data file created by the main batch file in the working directory. Files created: pot.out — Text file of output from *teapot*. This file may contain error messages produced by *teapot* when reading the data. cake_in.dat — Text file specifying the input data to *teacake* describing the FE structure in a similar style to the main data file. A node numbering system local to the FE domain is used. asm_in.tmp — Binary file containing information relating local node numbering for each domain to the global node numbers. This file also contains the specification of the nodal loads at the globally numbered nodes. bein_*n*.dat where *n* = 1, 2, .. — A text file is created for each BE domain. The data is specified in a style similar to the main data file and uses the local node numbering system for the domain. betop_*n*.plt where *n* = 1, 2, .. — A *Matlab* loadable text file containing the element connectivity data for BE domain *n* specified in the global node numbering system. globxy.plt — A *Matlab* loadable text file specifying the co-ordinates of all nodes by their global node number. fetop.plt — A *Matlab* loadable text file specifying the connectivity data for the finite elements using the global node numbering system. lodnod.plt — A *Matlab* loadable text file listing the nodes which have a load applied at them.

TEACAKE.exe	<u>Calculate and Assemble finite element matrices ($\mathbf{K}_e$ etc.)</u> This executable reads the finite element domain data file and calculates the mass, stiffness and damping matrices for the finite elements. Files read:		
		cake_in.dat	Text file specification of the FE domain structure using the domain-local node numbering system created by *teapot*.
	Files created:		
		cake.out	Text file containing messages for the user.
		fe5_mtx.tmp	Binary file containing mass, stiffness and damping matrices for the FE domain in band limited storage.
TEABAGS.exe	<u>calculate **B**oundary element matrices **A**ssemble and perform **G**lobal **S**olution</u> This executable reads the finite element matrix file and at each frequency of the analysis, calculates a dynamic stiffness matrix for each boundary element domain, assembles the FE and BE domain matrices and solves the global matrix equation. Files read:		
		fe5_mtx.tmp	The FE mass, stiffness and damping matrices
		bein_*n*.dat where n = 1, 2, ..	The input text files for each of the BE domains.
		asm_in.tmp	Global and domain node numbering correspondence data.
	Files created:		
		tea.out	Text file of output.
		hs_*n*.tmp where n = 1, 2, ..	Binary files of diagonal tensors of the static part of the H matrix for each BE domain. These contain the singular terms calculated once and reused for each frequency. The file stores these results for each BE domain while the matrices for other BE domains are being calculated.

3. DATA PREPARATION

The MSDOS batch program TEA is run from a command line specifying two parameters, namely the input data file name and the output file name (including path). TEA has a single input data file, the structure of which is described in this Section. An example data file is presented in Appendix A. The general method of data specification is the same as that used for the earlier single domain boundary element program *gbed4* [2].

The main input data file for TEA is described here. The sub-program *teapot* produces two other text data files that are of similar kind, firstly the data to set up a finite element sub-domain (i.e. comprising all the specified finite elements) and, secondly, to set up individual boundary element domains (each consisting of a single material). No specification of these data files is given here. Should the user wish to examine or modify these files directly, the data can be readily understood by analogy with the main data file.

3.1 General method of data specification

The data is specified in the input text file in *modules* under the following *headers*. A *header* must start with the '*' character but may be in upper or lower case. The following list of headers/modules are used in the main input data file.

**title*

**material*

**frequency*

**nodes*

**ansys*

**finite elements*

**boundary element domain*

**loads*

**end*

**

After the header, the relevant data must appear in the following lines without blank lines between lines of data. Except where otherwise stated, the data modules may be placed in any order. Some modules are obligatory, the appearance of other modules depending on the particular model to be set up.

The data for each module is described in the following sections 3.2 to 3.11. Numerical data is read line by line with no regard to column position except that it must be contained in the first 132 columns of the line in all cases. A line of numerical data may contain a *comment* after the required number of data items. The start of the comment is indicated by a '!' character. In general the program will check that the required number of numeric data items is found on the line and will halt, giving an error message in the output file, if these are not found.

SI units (*i.e.* in the base dimensions N, m, s) should be adhered to for input parameters in all data modules. The co-ordinate axes in the vertical plane of the model are x, from left to right, and y positive upwards.

The maximum number of nodes, elements, frequencies and field points, are set at the time of compiling the program in the include files *pot_dim.inc*, *cake_dim.inc* and *bag_dim.inc* which affect the executables *teapot.exe*, *teacake.exe* and *teabags.exe* respectively. These files are listed and other information for compiling the program may be found in Appendix B.

3.2 The **title* module

This module is optional. After the header only one line of data is required. This is a line of up to 132 characters of text. The text is used in the text data files written by *teapot* and appears in the main output file written by *teabags* in order to identify it. It is suggested that this input be used to identify analyses, giving date information, a structure name, job reference to keep track of data and output files.

3.3 The **material* module

Each data line after the header should contain the values of Young's modulus, *E*, the Poisson's ratio, the mass density and loss factor for a single material. At least one data line, *i.e.* the specification of one material is required. The maximum number of different materials that is allowed by the program may be adjusted using the parameters in the include files for array dimensions (see Appendix B). Each material is referenced in the finite elements specification and boundary element domain specification data according to the ordinal number in which it appears in the **materials* module. In models in which a large number of materials are referenced the user may wish to indicate this number along with a material description using comments after each line of data in order to ensure clear association of elements/BE domains with the appropriate material.

3.4 The **frequency* module

This module specifies the list of frequencies at which the displacement responses are required. The header is followed by a list of frequencies which should be typed one per line. The results will be output for each frequency in the same order in which they appear in this list.

3.5 The **nodes* module

The co-ordinates of the nodes are listed in this module. This module is obligatory and should only appear once in the data file. It is preferred, though not required, that it should appear before the **finite elements* and **boundary elements* modules. This allows greater checking of the validity of the data in these element specification modules.

A single line entry containing the x and the y co-ordinate must be made for each and every node of the model ('mesh'). No node numbers are specified as these are generated by the program. However, it has been found useful for cross-reference for the user to mark the node numbers after the nodal co-ordinates using the comment facility (see example data file in Appendix A). The ordinal number of the node as it appears in this data module is used as the *global node number*. The text data files written by *teapot* for calculating the finite element matrices in *teacake* and for each BE domain, used by *teabags,* define *local node numbers*. The loads and element topology specifications in the main data file use the global node numbers.

No redundant nodes are permitted, *i.e.* each node must belong to at least one finite element or boundary element.

3.6 The **ansys* module header

This header is optional. It appears on its own without any lines of data. If it is to be used it must be specified before the **boundary elements* modules or the **finite elements* module (if present). The action of this module header is to alter the ordering of the nodes for each element specification in both the **boundary elements* modules and the **finite elements* module to be compatible with that produced by the ANSYS pre-processor. If the **ansys* header is not present the default node ordering for TEA should be used. The alternative node numbering schemes are indicated in Sections 3.7 and 3.8.

3.7 The **finite elements* module

This module specifies the type and connectivity of each of the finite elements in the model. The module contains one line of data for each element of the FE mesh. As is the case for the nodal co-ordinate data, no numbering is specified to the program as this is generated by the program itself. Again it is useful for cross-referencing with node numbers for the user to mark the element numbers in order using the comment facility.

The structure of the data listed for each element is as follows. The data for a single element must occupy only a single line in the data file and must not exceed 132 columns in total.

- the first number of the line is an integer specifying the element type. The element types that are available are listed in Table 2 and illustrated along with their node numbering in Figure 2.

Table 2 Finite elements available in TEA.

Element description	Type no.	Order of approximation used
Four noded quadrilateral	1	Linear shape functions
Three-noded triangle	2	Linear shape functions
Eight noded quadrilateral	5	Quadratic shape functions
Six noded triangle	6	Quadratic shape functions

- the second number is an integer, *imat*, referencing the material parameters of the element as given in *imat*th line of the **materials* data module.
- The following items on the line should be integers specifying the node numbers which belong to the finite element. The number of these should correspond to the number of nodes in the element type. None of the nodes of the element may be omitted and it is important that the order in which they are given is adhered to. The order of the nodes depends upon the presence of the **ansys* module. The ordering is specified in Figure 2. Where the *ansys module has been placed before the **finite element* module the node numbers are specified in the order indicated by i, j, k, ... otherwise they follow the order indicated by 1, 2, 3 ....

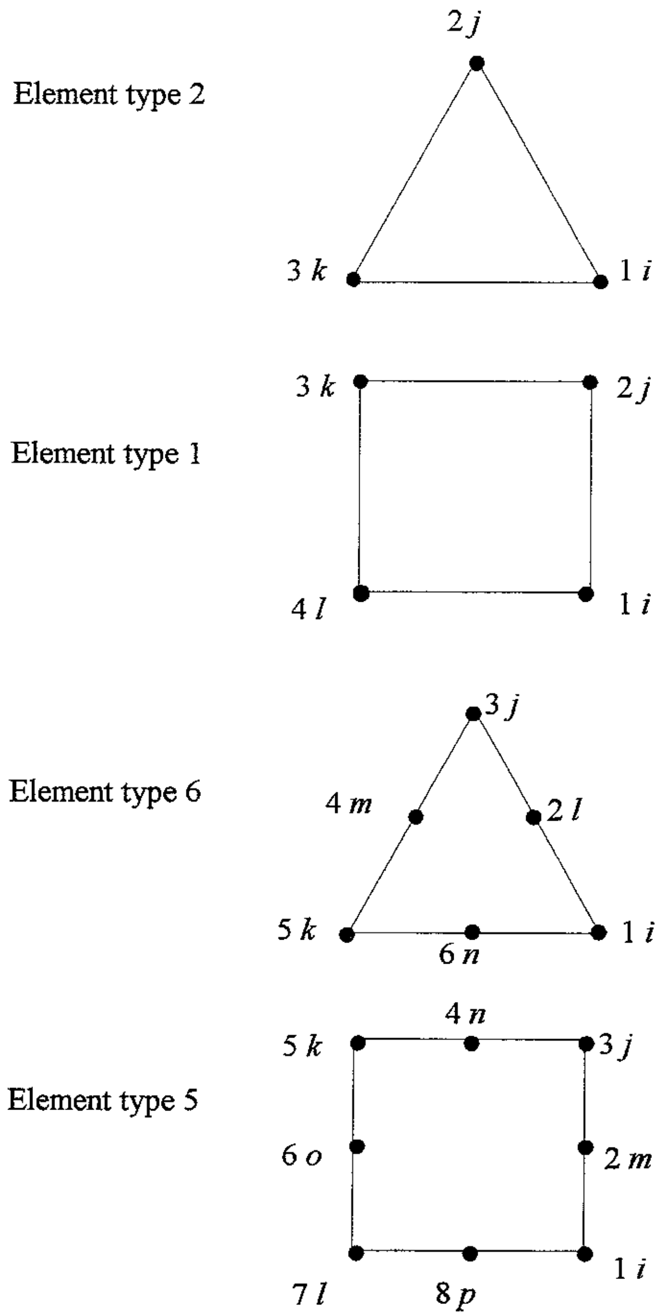


Figure 2. Finite element node numbering.

3.8 The **boundary elements* module

A **boundary elements* module should appear in the data for each boundary element sub-domain of the model. The number of BE domains that may be used is controlled by an array dimension parameter in the include files (see Appendix B).

Each BE sub-domain consists of a single material only. The reference number for this material (its line number within the **materials* module) is given as the first line of data in the **boundary elements* module. The subsequent lines of the **boundary elements* module specify the nodes of each boundary element of the sub-domain. There is only one type of boundary element available, namely a three-noded element. A single line of three integers, the global node numbers of the element, must be given to specify each element of the domain. The order in which the node numbers are given depends on whether the **ansys* module header is used. If the **ansys* header has been placed in the data file before the **boundary elements* module, the nodes are specified in the order indicated by the letters *i, j, k* in Figure 3 otherwise they are specified as indicated by the numbers 1, 2, 3 shown in the figure.

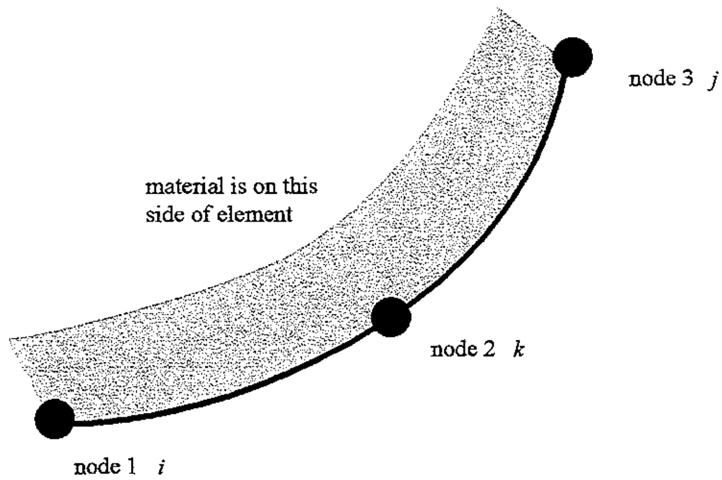


Figure 3. Three-noded, quadratic shape function boundary element.

3.9 The **loads* module

This module specifies the applied nodal forces. This module must occur once in the input data. Each line specifies the node number in the global node numbering system at which a nodal force is applied. The force is specified in the next four numbers which are read as floating point values. These represent the real and imaginary parts of the complex amplitude of force applied at the node in the x and y directions respectively. The same amplitude is applied at all frequencies. Only nodes at which a non-zero load is applied should be mentioned in the module.

Loads may be specified on any of the nodes whether part of an FE or BE sub-domain. In order to apply a distributed load (pressure amplitude) over the three nodes of one side of an 8-noded quadrilateral or 6-noded triangle, or over a 3-noded boundary element, the nodal forces making up the total force applied on that element should be applied at the nodes in the ratios 1/6, 2/3, 1/6 as is usual in the use of the finite element method (e.g. page 517 of reference [6]).

Note that the software does not provide for the application of prescribed displacements or constraints as it is intended for use in modelling ground vibration.

3.10 The **end* header

This header signifies the end of the input data and should be placed as the last line of the input data file. There are no data lines associated with this header.

3.11 The ** header

This is a means of including comment lines between other data modules. All text on the rest of a line which starts with ** is ignored by the program. Comments included in this way must not interrupt the data in any of the other modules. There are no data lines associated with this header. There may be as many occurrences of this header as the user wishes.

3.12 Checking of input data and plotting the structure

A Matlab (version 4.2) program, called *teacups.m* (Check User data and Plot Structure), has been written for checking the input data and presenting it graphically. *Teacups* must be run from within Matlab. The program requires the name of the input data file (through a windows dialogue box). The program executes *teapots* so that the **.plt* files are created from the global data. By this means the data is tested by the reading and checking routines of the main software. A menu offers the choice of plotting the nodes of the model, the elements or the elements with normal vectors for the boundary element. The boundary element domains and the finite elements are represented in different colours. The node and element numbers can be added to the plot as well as a symbol indicating the nodes at which loads are applied.

If the plotting routines fail because a **.plt* file has not been created the user should examine the text file *pot.out* for error messages concerning the data file.

3.13 Results file

A single output file is written by the program TEA. This is a text file which echoes the input parameters and then lists the output at each frequency separately for all the *nodes*. The output is given on each line as the complex amplitude of displacement at each node location. It is listed in the following columns

- x co-ordinate of node or field point
- y co-ordinate of node or field point
- real part of complex amplitude of displacement in the x direction
- imaginary part of complex amplitude of displacement in the x direction
- real part of complex amplitude of displacement in the y direction
- imaginary part of complex amplitude of displacement in the y direction

Note that, although the output is listed in the order of the global node numbers used inside the program, no node numbers are referenced in the output file.

For plotting the results at a single frequency as a function of the nodal co-ordinates, the output file is easily edited into a form which can then be read into a spreadsheet or "loaded" into Matlab. An ancillary program, *ex_frf*, is provided to extract frequency response functions from the output file at a single node location.

4. EXAMPLE RESULTS

4.1 Parameters for the examples

In order to validate the program and to investigate the accuracy of the method a number of examples have been analysed. These demonstrate the capability of the program to model an elastic half-space and a layered ground as a relatively thin soil layer overlying a stiffer half-space substratum. The examples all use a particular set of material properties which are given in Table 3. These parameters represent a relatively soft soil and substratum.

Table 3. Material properties for the soil.

	Top layer or half-space	Substratum half-space
Young's modulus	157×10^6 Nm^{-2}	1062.8×10^6 Nm^{-2}
Poisson's ratio	0.18	0.253
Density	1517 kgm^{-3}	1759 kgm^{-3}
Loss factor	0.1	0.1
Compression wave speed	336 ms^{-1}	854 ms^{-1}
Shear wave speed	210 ms^{-1}	491 ms^{-1}

All the analyses presented here are run on a PC well within its memory capacity of 128 Mb.

4.2 Homogeneous half-space

For the purpose of comparison with results from the semi-analytical model and the results presented in [2], a homogeneous elastic half-space with the same material parameters as used in [2] has been analysed. The half-space has been made to include a 6 m wide by 2 m high block of soil modelled using finite elements. This block has the same parameters as the surrounding medium and has been included in order to verify the process of combining a finite element domain with a boundary element domain. The finite element block occupies the region from x = -3 m to +3 m and a load of width 3 m is distributed along the nodes at the surface (y = 0 m) from x = -1.5 m to +1.5 m. The total load is 1 N so that the results represent frequency response functions. The surface of the ground is modelled by boundary elements out to -10 m and out to + 30 m along the line y = 0 m. The model mesh is shown in Figure 4.

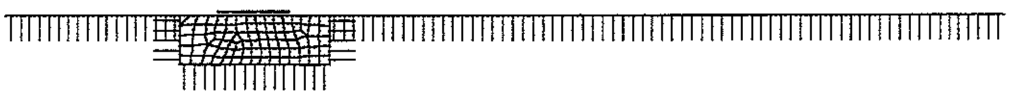


Figure 4. The structure for the basic model for a homogeneous half-space showing normal vectors pointing into the BE domain material.

Results from this model are directly comparable to those from a two-dimensional semi-analytical model [4]. The semi-analytical model has been used to provide a comparison by which the approximation error in the coupled FE/BE approach may be evaluated. It should be noted however that the semi-analytical model does use an adaptive numerical integration process in its calculation that may introduce some calculation error for high frequencies and very low response levels.

A study of the error incurred in the BE approach where the BE matrices are not transformed in terms of equivalent nodal forces and the loads are applied as tractions has already been made in [2]. The effects of the termination of the surface on the results near these points have been examined in [2] and are known to be confined to a limited distance from the last node of the boundary. These characteristics of the boundary element model are not discussed again in the present report. Although the situation for wave propagation is symmetric about the centre of the load the surface of the ground has not been modelled to the same distance on the left hand side and on the right in order to reduce the number of elements and hence to improve the efficiency of the calculation. Only the results from the right-hand side are compared to those of the semi-analytical model which has the symmetry taken into account.

The lateral response along the surface at 50 Hz is shown in Figure 5 and the vertical response in Figure 6. The extent of the finite element block is indicated on these graphs. It can be seen that along the surface the model is quite accurate in comparison to the semi-analytical model for both the lateral and vertical response. In most of the distance range the dotted line for the semi-analytical model results cannot be distinguished from the solid line for the BE results. The exception is for the expected behaviour near the termination of the boundary at 30 m.

Figures 7 and 8 show the results for the lateral and vertical responses at 100 Hz. Here, although a slight increase in the error is observable, the results are also accurate for most of the distance range. The results for 200 Hz, Figures 9 and 10, show a clear degree of error emerging at all points along $y = 0$ m as the wavelength is now only about 2.5 times the element size. This reflects a similar limit of the number of elements per shear wavelength for accurate results as was found for the pure boundary element calculation [2]. At distances beyond 20 m the boundary element solution exhibits a 'noise floor' as the true response becomes more than a factor of 10^3 below the response near to the load. This has been explained in [2] as being due to the nature of the boundary element method in linking all degrees of freedom with all the others (reflected in the fact that full matrices are produced) and so the error, due to the size of the elements, associated with the parts of the model where there is a high level of response contaminates the low response levels well away from the load. The level of this error is a little higher in the FE/BE model than for the pure BE model [2], especially for the vertical response results.

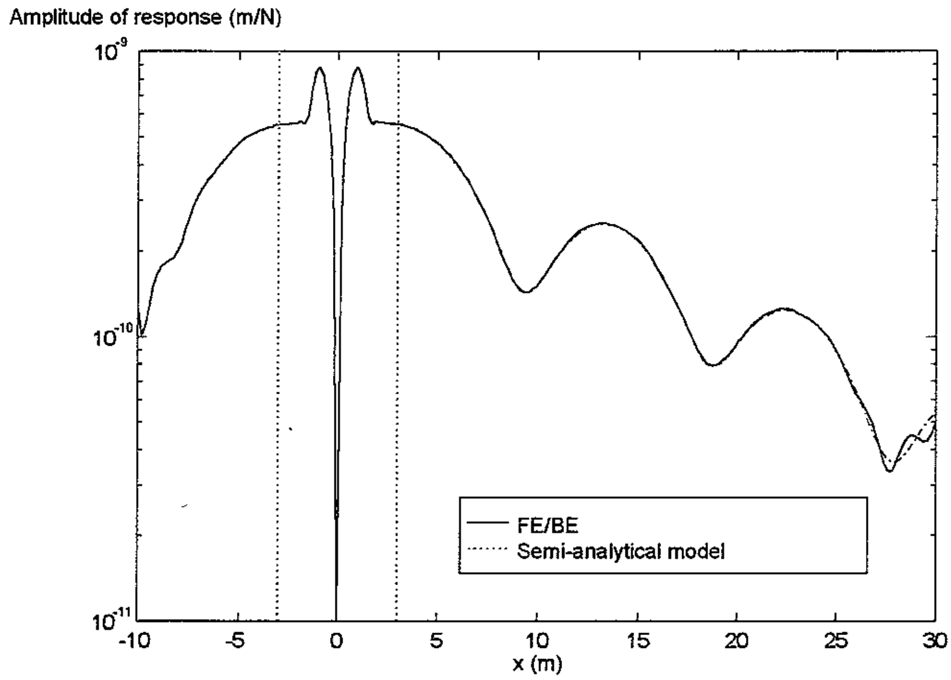


Figure 5. Lateral response at 50 Hz calculated using the combined FE/BE model and using the semi-analytical model.

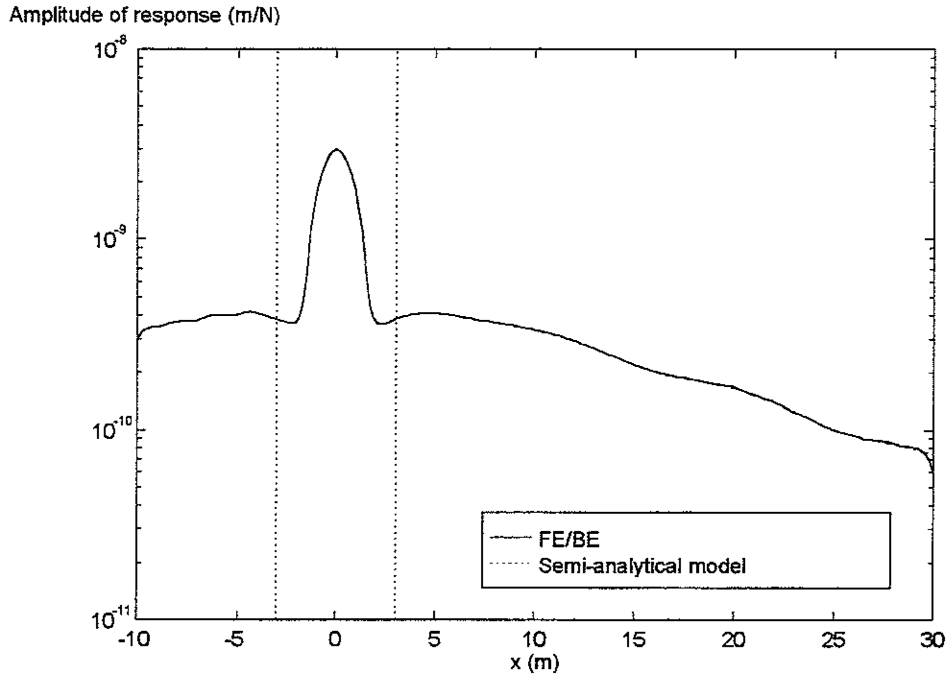


Figure 6. Vertical response at 50 Hz calculated using the combined FE/BE model and using the semi-analytical model.

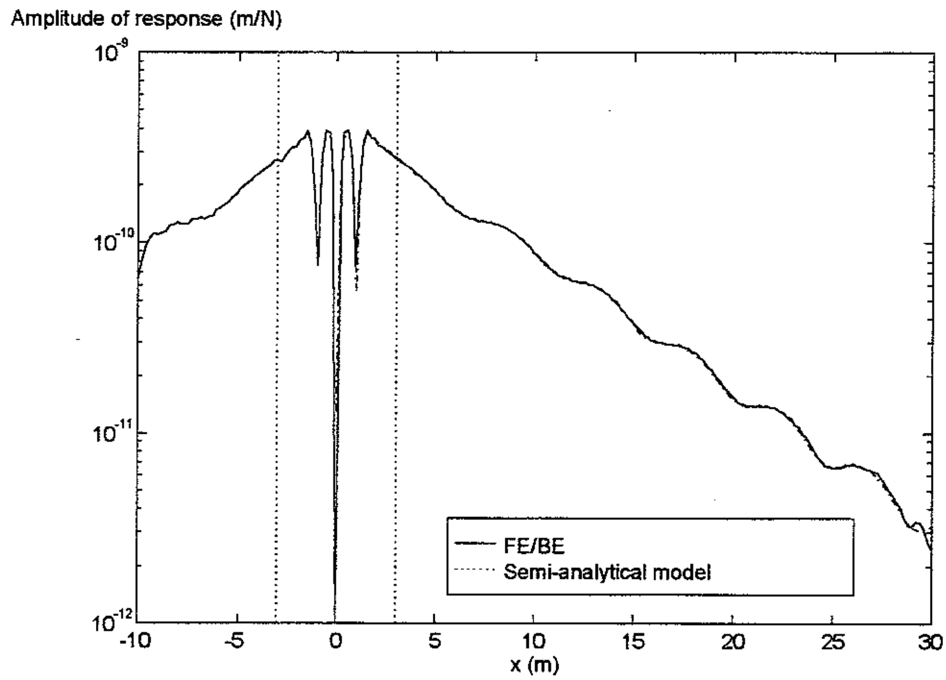


Figure 7. Lateral response at 100 Hz calculated using the combined FE/BE model and using the semi-analytical model.

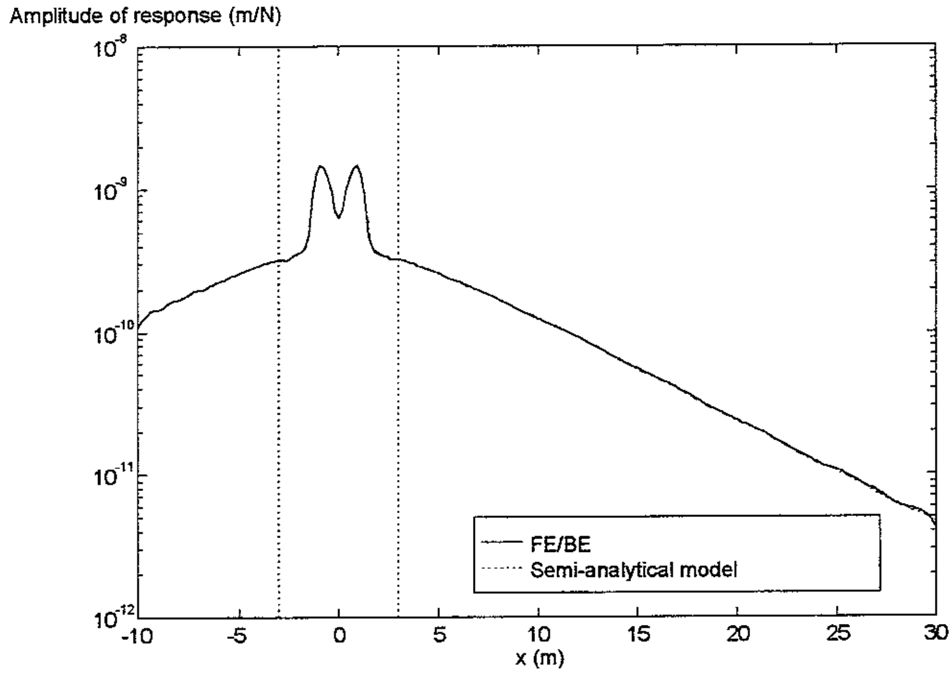


Figure 8. Vertical response at 100 Hz calculated using the combined FE/BE model and using the semi-analytical model.

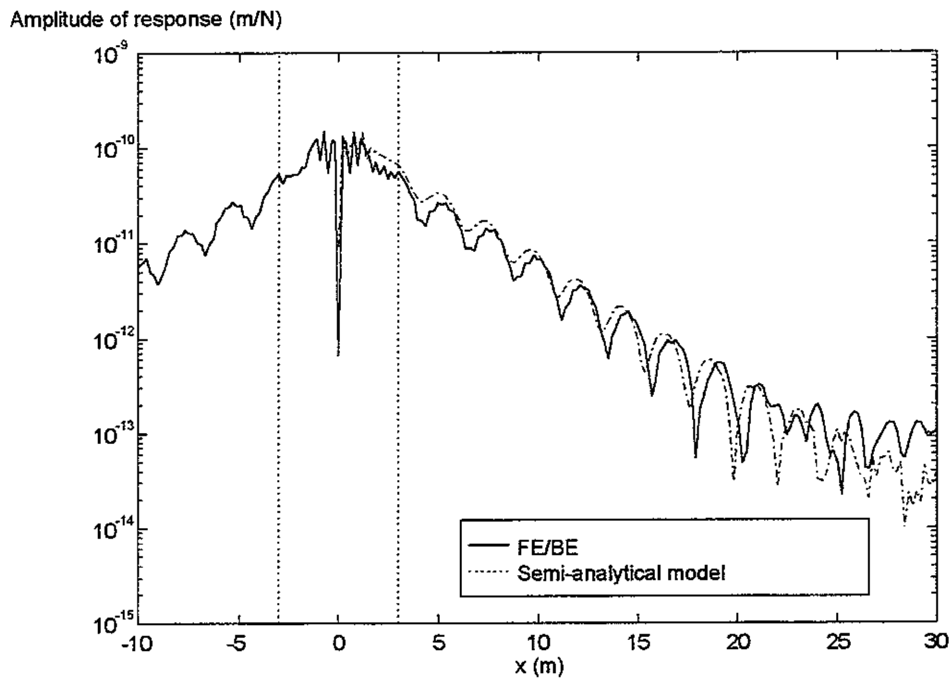


Figure 9. Lateral response at 200 Hz calculated using the combined FE/BE model and using the semi-analytical model.

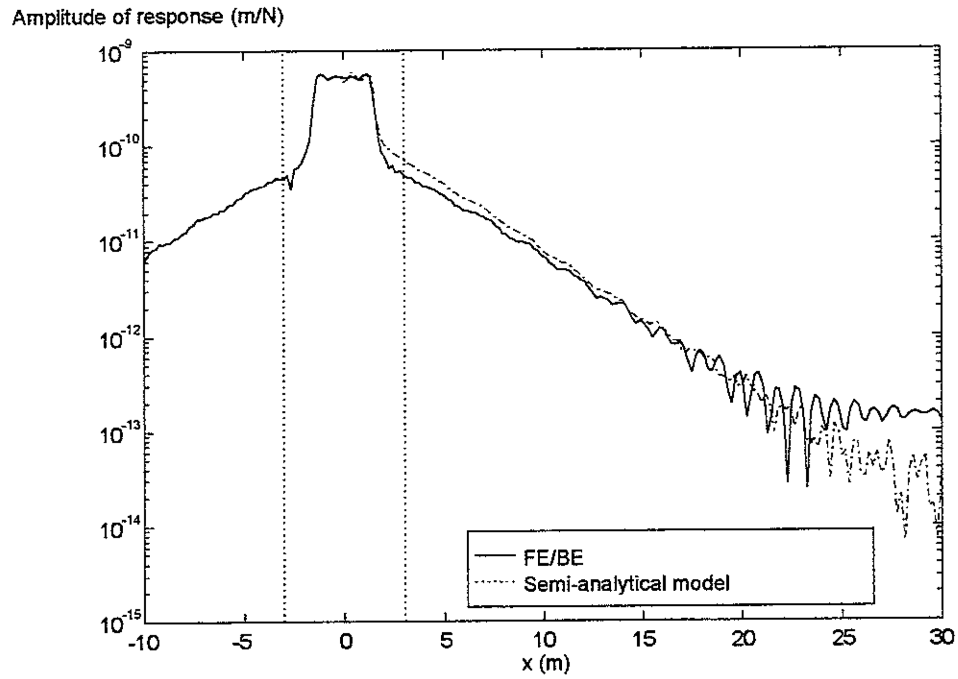
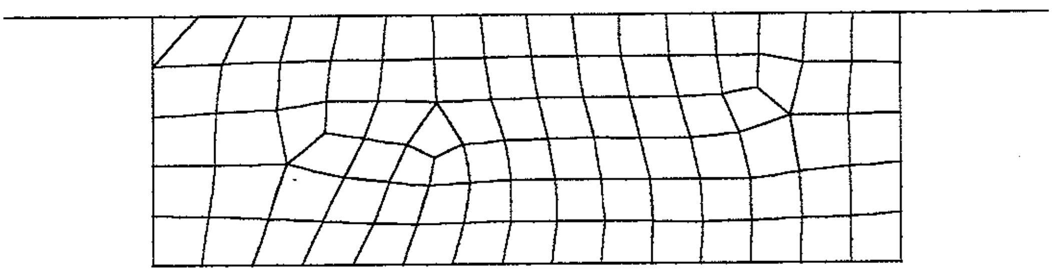
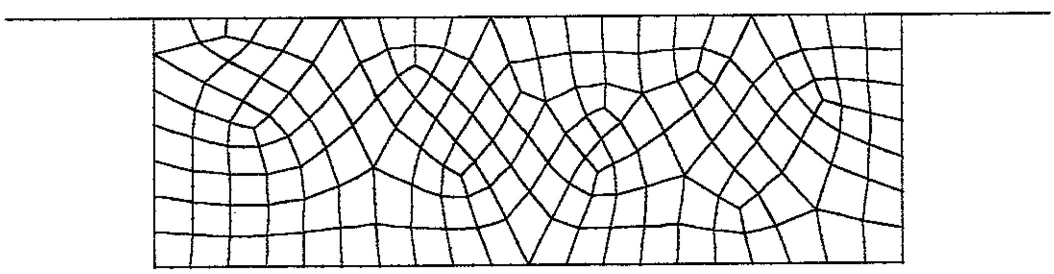


Figure 10. Vertical response at 200 Hz calculated using the combined FE/BE model and using the semi-analytical model.

From Figure 9, especially, it is noticeable that much of the error between the semi-analytical result and the FE/BE model is already committed within the FE block close to the source. In order to investigate this, a finer element mesh was defined in this part of the model. Figure 11 shows the initial and the refined finite element mesh. Figures 12 and 13 show the calculated response at 200 Hz for the model with the more finely meshed FE block. The boundary elements are also smaller round the FE block in order for them to match but away from the FE block they have been kept at the same length of 0.4 m. Only the 200 Hz results are examined as it has been established that lower frequency results do not exhibit a readily examinable degree of error.



(a)



(b)

Figure 11. part of the model showing the basic (a) and refined (b) FE meshes.

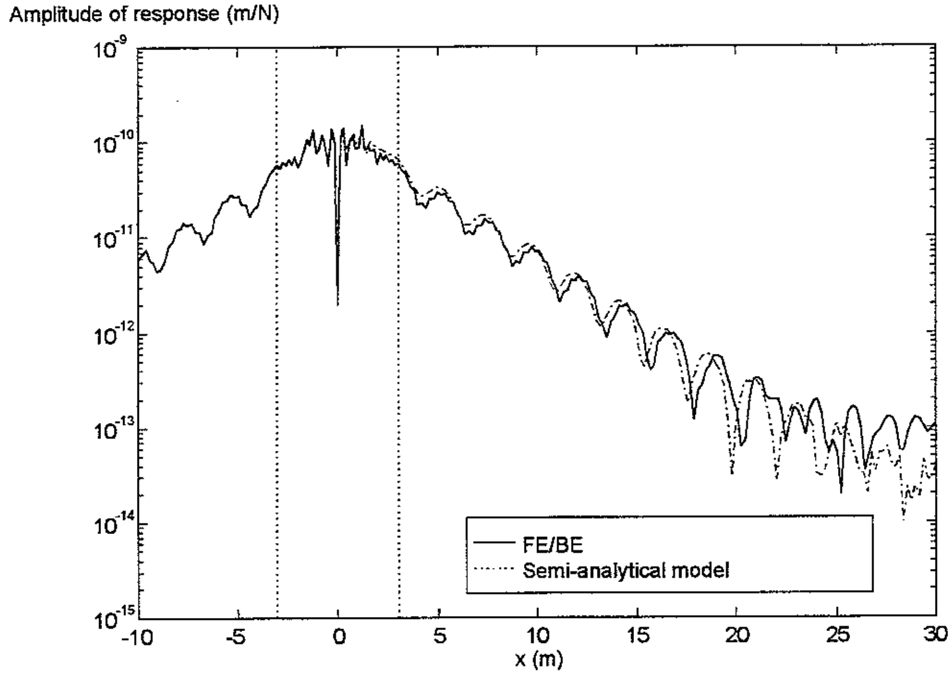


Figure 12. Lateral response at 200 Hz calculated using the combined FE/BE model with refined FE mesh and using the semi-analytical model.

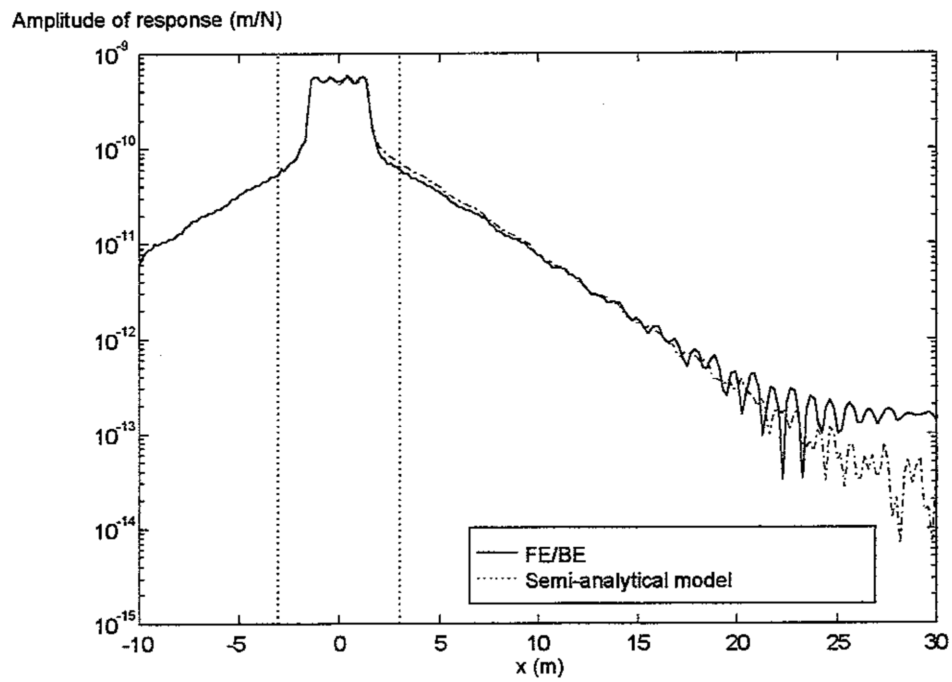


Figure 13. Vertical response at 200 Hz calculated using the combined FE/BE model with refined FE mesh and using the semi-analytical model.

A comparison between Figures 9 and 10 and Figures 12 and 13 shows that the refinement of the FE region results in a reduction of the error near to the load.

A further analysis was carried out for which the FE block was replaced by a second BE domain with the same outline geometry. The material of the second BE domain is still the same as that of the main BE domain so that the model is still of a homogeneous half-space. Figure 14 presents the mesh for this case with the normal vectors to the BE surfaces omitted. The size of the elements for the BE block are based on the geometry of the original FE block, i.e. they are bigger than those used to match the refined FE block.

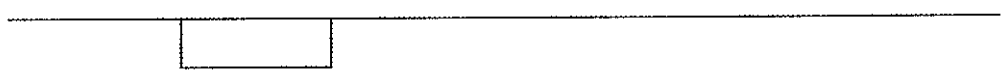


Figure 14. The model using a second boundary element sub-domain in the place of the FE block.

The results at 200 Hz for the BE block, the refined FE block and the semi-analytical model for the half-space are shown for the lateral and vertical responses in Figures 15 and 16. It can be seen that the results obtained using the BE block are even closer to the semi-analytical model results than the refined FE block. At the edge of the model the results are indistinguishable whether the FE or BE block has been used. It can be concluded then, that a block of boundary elements, when used to model the soil is more accurate than finite elements of similar or even smaller size, even though the block is finite in extent. This should be borne in mind when designing the idealisation for an analysis where it may be permissible, in parts of the model, to use either a BE or an FE dicretization.

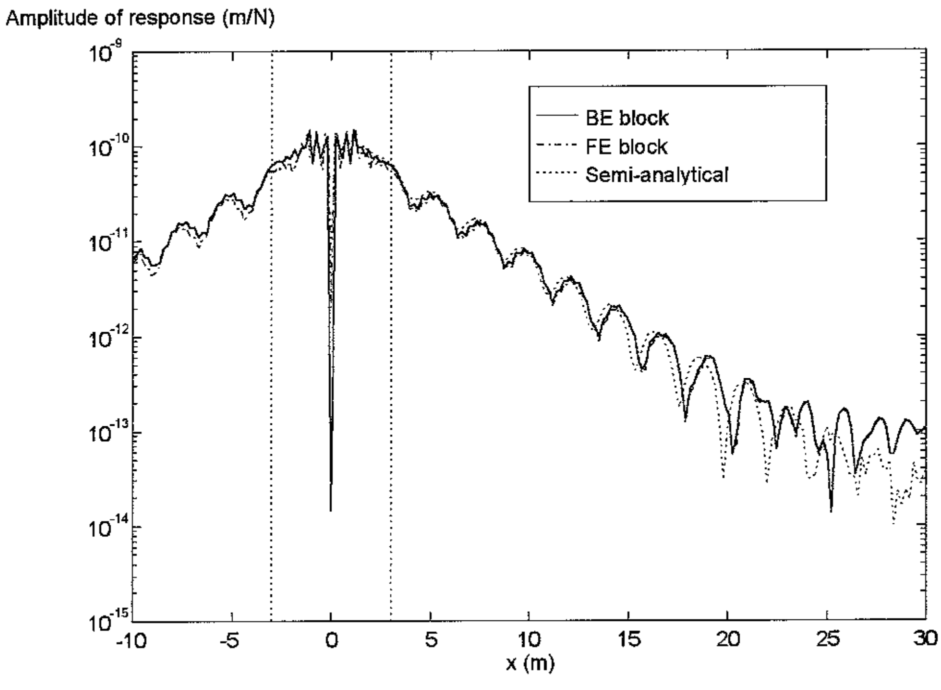


Figure 15. Lateral response at 200 Hz calculated using the BE only model with two subdomains, the refined FE block and using the semi-analytical model.

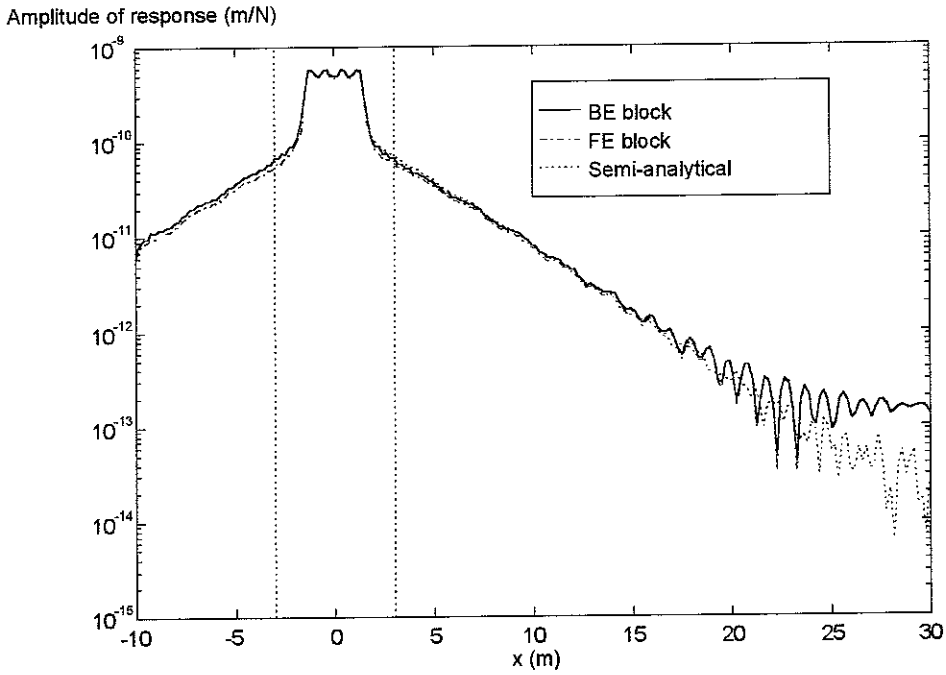


Figure 16. Vertical response at 200 Hz calculated using the BE only model with two subdomains, the refined FE block and using the semi-analytical model.

Finally, for the half-space, the results at 200 Hz have been obtained for a model of the whole half-space using only a single domain of boundary elements; *i.e* to form a straight line between $x = 30$ m and $x = -10$ m. These are compared with the semi-analytical model and the pure BE model [2], in which the BE matrices are not converted into an equivalent FE matrix, in Figures 17 and 18. The results are almost indistinguishable for the two BE methods. The conversion of the BE matrices to equivalent FE matrices does not therefore introduce significant error. Nor does the setting of the loads as distributed nodal forces make any difference compared to setting them as discontinuous vertical applied stress on the boundary element discretization [2].

It can be seen in Figure 18 that the general level of error where the response becomes small, beyond $x = 20$ m for the equivalent FE matrix method is the same as that incurred by the pure BE method, being lower than that encountered in the models with the 2 by 6 m block. This suggests that the origin of the increase of this kind of error for the combined FE/BE model lies in the linking of the domains with arbitrary (right-angled) geometry. (Recall that Figures 14 and 15 show that this error is the same whether a BE, FE or finer FE block is used).

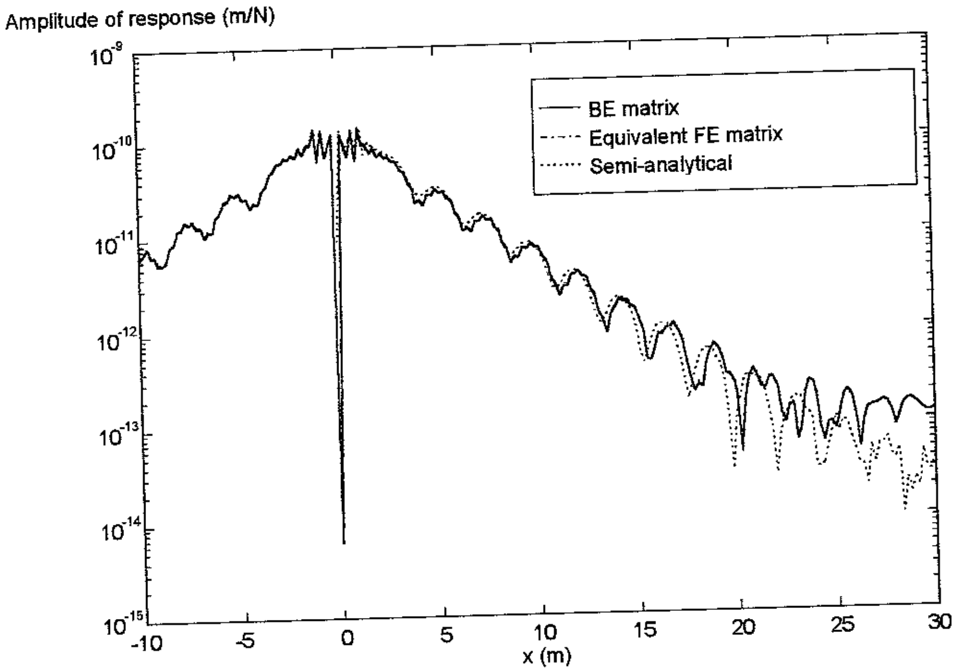


Figure 17. Lateral response at 200 Hz calculated using BE's in two ways.

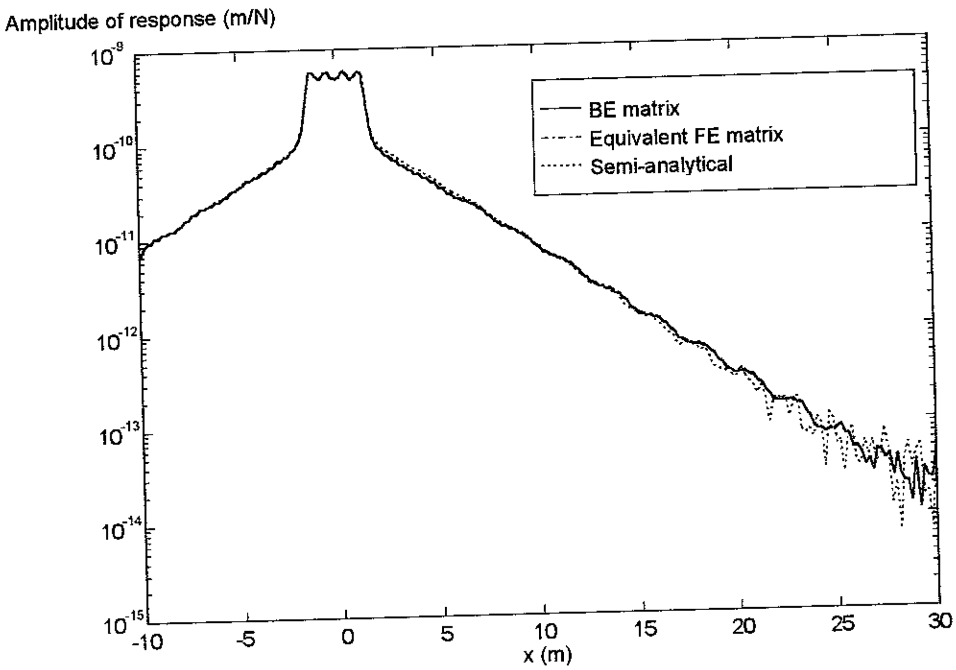


Figure 18. Vertical response at 200 Hz calculated using BE's in two ways.

4.3 Layered half-space

In order to confirm that the model can be used to examine the behaviour of layered soils a model in which a 2 m thick, 'weathered' layer overlies a stiffer half-space of material has been constructed. This model uses the same FE block as the first half-space model presented, *i.e.* the coarser mesh, and three boundary element domains. The model is shown in Figure 19. The layer (2 BE domains and the FE block) have the softer material properties indicated in Table 3 (previously used for the whole half-space). The substratum is attributed the stiffer properties indicated in Table 3.

Figure 19. Model of ground with layer overlying a stiffer half-space of material and using the original 6 m wide 2 m deep block of finite elements (unit normal vectors are shown pointing into the material of the BE domains on this model).

The results for the layered ground for the lateral and vertical responses at 50 Hz are shown in Figures 20 and 21. These are compared with semi-analytical results based on the model in [4]. As for the constrained 2 m thick layer in reference [2], a greater effect of the termination of the boundary is encountered than for the half-space case. Apart from this, the results are again accurate for this frequency.

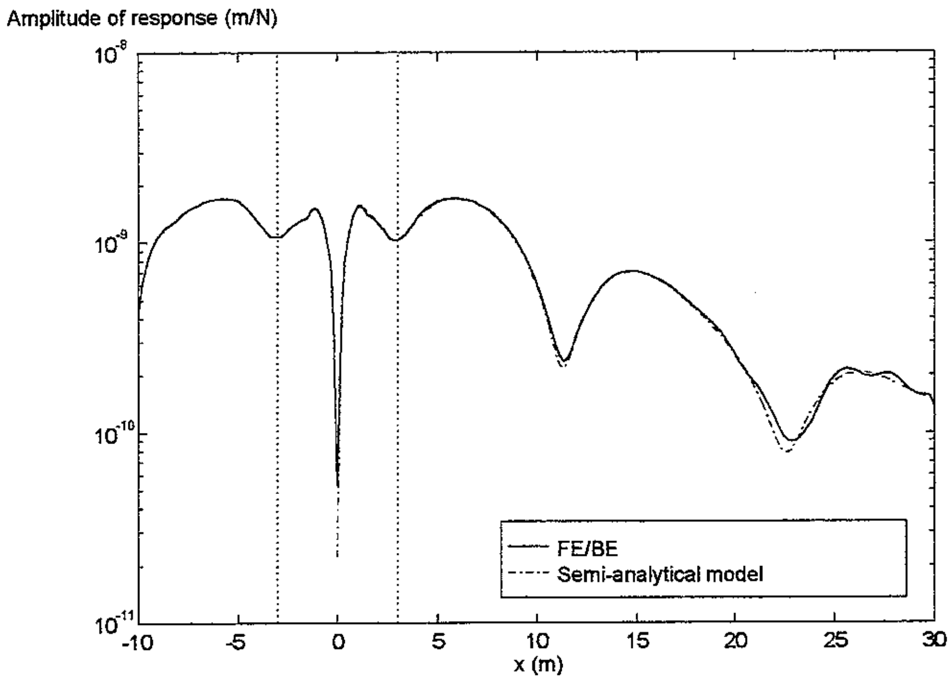


Figure 20. Lateral response of layered ground at 50 Hz calculated using model with FE block and three BE domains compared to results from the semi-analytical model.

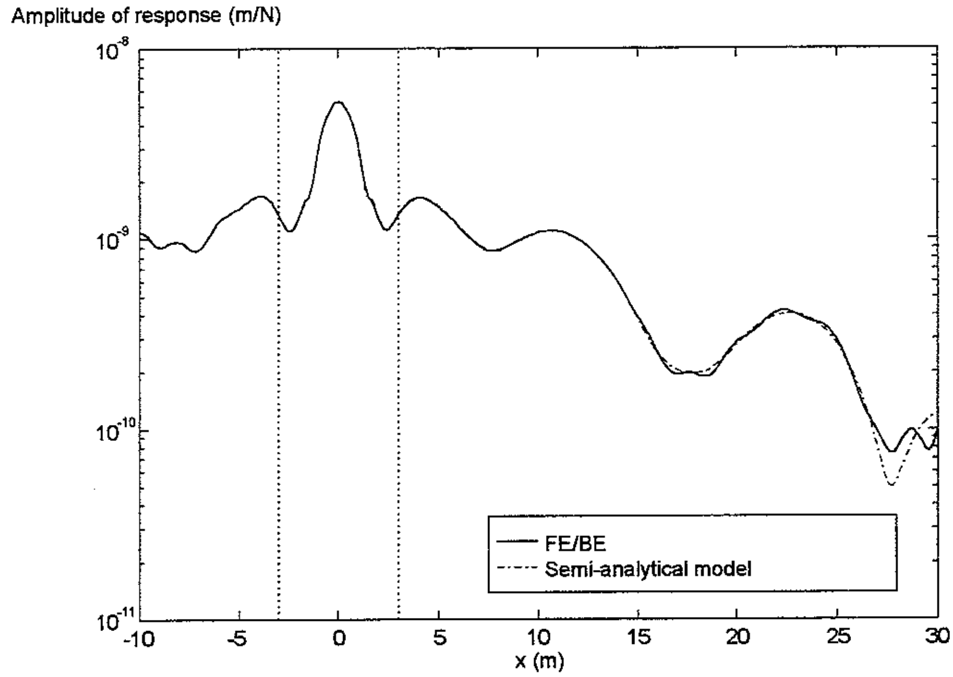


Figure 21. Vertical response of layered ground at 50 Hz calculated using model with FE block and three BE domains compared to results from the semi-analytical model.

The lateral and vertical responses at 100 Hz are presented in Figures 22 and 23. The more complicated nature of the response as a function of distance along the surface for the layered ground, than for the half-space can be seen in these figures. This is due to the modal nature of propagation in the layered ground [2]. However, it can be seen, in the comparison with the semi-analytical model, that a good degree of accuracy is maintained.

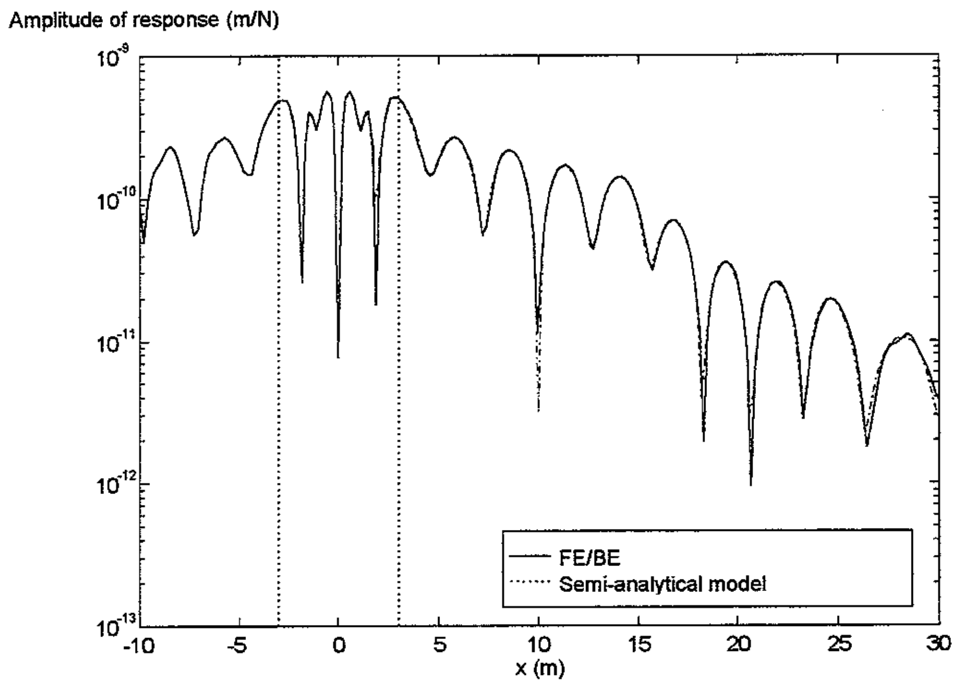


Figure 22. Lateral response of layered ground at 100 Hz calculated using model with FE block and three BE domains compared to results from the semi-analytical model.

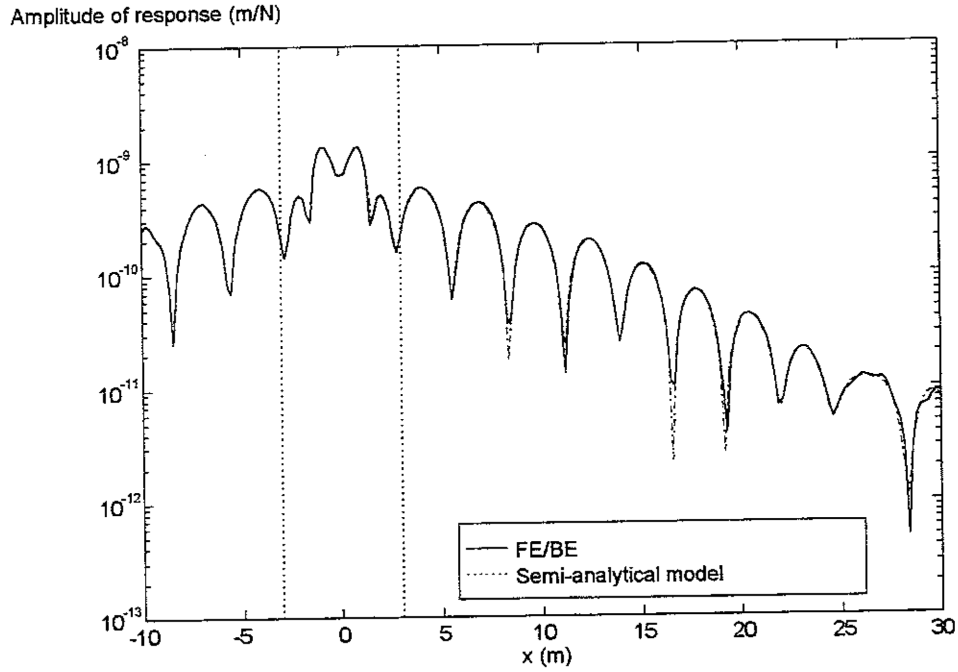


Figure 23. Vertical response of layered ground at 100 Hz calculated using model with FE block and three BE domains compared to results from the semi-analytical model.

Figures 24 and 25 present the lateral and vertical surface responses for the layered ground model at 200 Hz. Although the function of amplitude of response with distance along the surface is now a very much more complicated shape, with sharp dips in the response, compared to the half-space case, the degree of accuracy is similar to that shown in Figures 9 and 10. Note that, in this case, there is little evidence of the 'noise floor' in the low levels of response at the edges of the model as the attenuation with distance in the layered ground is less than in the half-space.

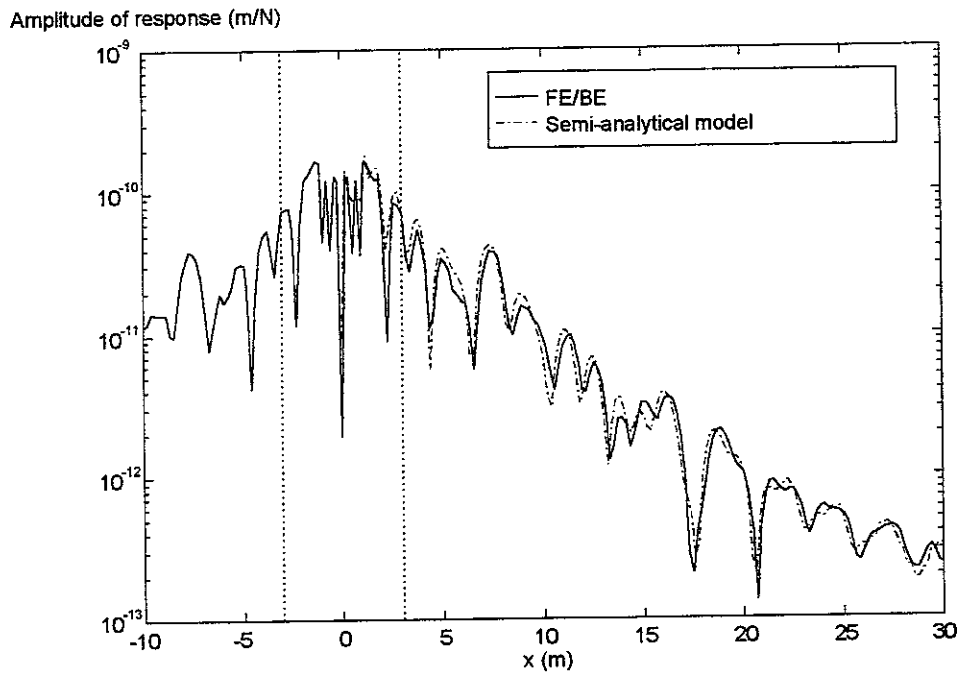


Figure 24. Lateral response of layered ground at 200 Hz calculated using model with FE block and three BE domains compared to results from the semi-analytical model.

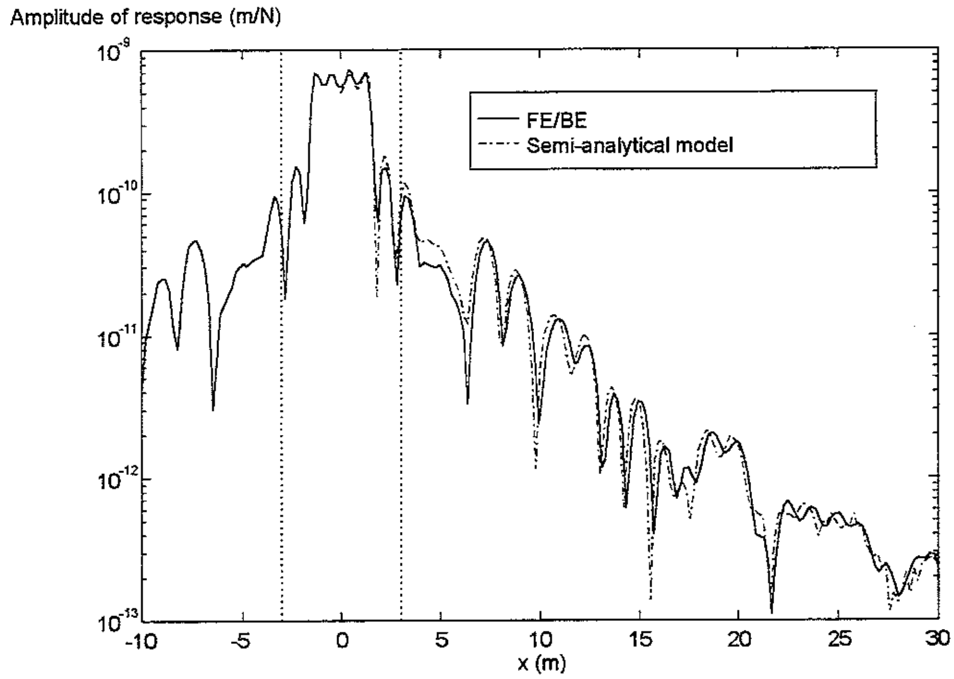


Figure 25. Vertical response of layered ground at 200 Hz calculated using model with FE block and three BE domains compared to results from the semi-analytical model.

5. CONCLUSIONS

A computer program has been produced that combines the ability to model wave propagation in an infinite medium using the boundary element method with the ability to model thin, detailed, inhomogeneous finite structures, embedded in, or in contact with, the ground, using the finite element method.

The program implements both linear and quadratic triangle and quadrilateral finite elements. For the boundary element subdomains a quadratic, three-noded element has been implemented. Tests on the model for a half-space and a layered half-space show that the degree of error for realistic soils for modelling in the frequency range up to 200 Hz is acceptable using a model that will run within 128 Mb of memory space on a PC.

When using the model the element size should be chosen so that there are at least 6 nodes (or two and a half elements) per shear wavelength in the soil for accurate results (corresponding to the 200 Hz results presented). Some error may be expected at locations where the response is more than 3 decades smaller than the response near the source. Results near to the termination of boundaries are subject to error. The boundary should therefore be extended well beyond the distances at which results are required.

A parallel report [3] has been produced which describes the theory implemented in the model. That report also presents the results of analyses comparing the vibration propagated from a lined and unlined tunnel structure.

6. ACKNOWLEDGEMENT

The work reported has been carried out, and the program described in this report has been developed, under EPSRC grant number GR/L11397.

7. REFERENCES

1. Von Estorff, O. and Kausel, E., 1989 *Earthquake engineering and structural dynamics*, **18**, 1065 - 1075, Coupling of boundary and finite elements for soil-structure interaction problems.

2. Jones, C.J.C., Thompson, D.J. 1999 *ISVR Technical Memorandum No. 838,* A boundary element model for two-dimensional elastodynamics on a single open or closed domain.

3. Jones, C.J.C., Thompson, D.J. and Petyt, M., 1999 *ISVR Technical Memorandum in preparation,* Two dimensional elastodynamic analysis using the combined boundary and finite element methods.

4. Jones, D.V. and Petyt, M. 1993 *Journal of Sound and Vibration* **161**, 1-18. Ground vibration in the vicinity of a strip load: an elastic layer on half-space .

5. Dominguez, J. 1993 *Computational Mechanics Publications, Southampton (Elsevier Applied Science.)* Boundary elements in dynamics.

6. Petyt, M. 1990 *Introduction to finite element vibration analysis,* Cambridge University Press.

<u>Addendum</u>

Reference 3 produced as

Jones, C.J.C., Thompson, D.J. and Petyt, M. Ground-borne vibration and noise from trains: elastodynamic analysis using the combined boundary element and finite element methods. *ISVR Technical Memorandum,* 1999, 844.

Appendix A Example data file.

This appendix lists a simple example data file consisting of a single finite element and three boundary elements forming an open boundary.

```
*title
Simple case with one FE and 3 BE
*material
500d6,0.3,2300.0,0.01   ! E, nu, rho, loss factor for FE
157.2d6,0.18,1517.0,0.1 ! E, nu, rho, loss factor for BE
*frequency
     4.00
*nodes
    1.50000000000         0.00000000000    !       1
    1.00000000000         0.00000000000    !       2
    0.50000000000         0.00000000000    !       3
    0.00000000000         0.00000000000    !       4
    -0.5000000000         0.00000000000    !       5
    -1.0000000000         0.00000000000    !       6
    -1.5000000000         0.00000000000    !       7
    0.50000000000         0.50000000000    !       8
    -0.5000000000         0.50000000000    !       9
    0.50000000000         1.00000000000    !      10
    0.00000000000         1.00000000000    !      11
    -0.5000000000         1.00000000000    !      12
*finite elements
    5    1    3    8    10    11    12    9    5    4  !     1
*boundary element domain ! number 1
 2                  ! material number
 1    2    3        ! element topology data
 3    4    5
 5    6    7
*loads
11  0.0 0.0 1.0 0.0 ! single point load
*end
```

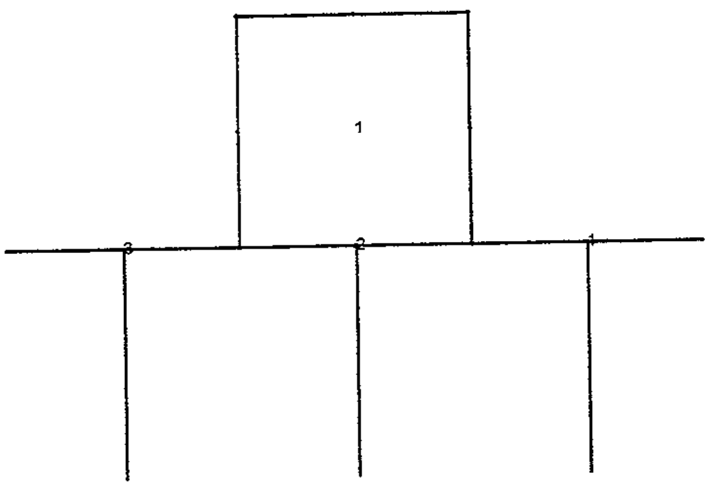


Figure A.1 Diagram of model produced using teacups showing element numbers and unit normal vectors facing into material of BE domain.

Appendix B Include files that dimension array space in *teapot*, *teacake* and *teabags*.

Include file *bag_dim.inc* for compilation as part of *teabags.for*.

```
c     fortran include file to be compiled with teabags.for
c     CJCJ ISVR 21-5-99
c
c     Parameters that limit the size of global and FE mesh etc
c
c     dimelm      number of elements in FE mesh
c     dimmat      number of material properties
c     dimbnd      semi-bandwidth of FE mesh
c     dimnod      global number of nodes
c     dimpnt      number of point stiffness elements allowed
c
      integer     dimelm,dimmat,dimbnd,dimnod,dimpnt,dimld
      parameter   (dimelm=400,dimbnd=300,dimnod=1050,dimld=100)
      integer     dimfrq
      parameter   (dimfrq=200)
c
c
c      Sets array dimension parameters in each subroutine in gbed6 subroutine
c
c      NNE= maximum number of boundary elements in a single domain
c      NX=  maximum size of BE equation system
c           which is twice the maximum number of nodes
c
      integer nne,nx,dimbed
      parameter(nne=250,nx=1000,dimbed=6) ! nx=1000 is > 40Mb in BE system
c
c     dimK = maximum size of coupled FE/BE system dynamic stiffness matrix
c
      integer dimK
      parameter(dimK=2100) ! this is 40Mb on its own
```

Include file *pot_dim.inc* for compilation as part of *teapot.for*.

```
c     fortran include file to be compiled with teapot.for
c     cjcj 18-5-99
c
c     Parameters that limit the size of global and FE mesh etc
c
c     dimelm      number of elements in FE mesh
c     dimmat      number of material properties
c     dimbnd      semi-bandwidth of FE mesh
c     dimnod      global number of nodes
c     dimpnt      number of point stiffness elements allowed
c
      integer     dimelm,dimmat,dimbnd,dimnod,dimpnt,dimld
      parameter   (dimelm=400,dimmat=40,dimbnd=300,dimnod=1050,
     *             dimpnt=40,dimld=100)
      integer     dimfrq
      parameter   (dimfrq=200)
c
c
c      Sets array dimension parameters in each subroutine in gbed6 subroutine
c
C      SET MAXIMUM DIMENSIONS
C      NNE= MAXIMUM NUMBER OF ELEMENTS
C      NX=  MAXIMUM DIMENSION OF THE SYSTEM OF EQUATIONS
```

```
C       NX=  2*MAXIMUM NUMBER OF NODES >= approx 4*MAXIMUM NUMBER OF ELEMENTS
C       NFM=MAXIMUM NUMBER OF FREQUENCIES
C
      integer nne,nx,dimbed
      PARAMETER(NNE=250,NX=1000,dimbed=6)
```

Include file *cke_dim.inc* for compilation as part of *teacake.for*.

```
c     This file to be used with code teacake.for
c     sets size of arrays for limit of size of model
c     CJCJ ISVR 16-4-99
c
c     Parameters that limit the size of mesh etc
c
c     dimelm      number of elements in mesh
c     dimmat      number of material properties
c     dimbnd      semi-bandwidth of mesh
c     dimnod      number of nodes
c     tdimnod     number of modes*2
c     dimpnt      number of point stiffness elements allowed
c
      integer     dimelm,dimmat,dimbnd,dimnod,tdimnod,dimpnt
      parameter   (dimelm=400,dimmat=40,dimbnd=300,dimnod=1050,
     *             tdimnod=2100,dimpnt=40)
```