

3-28-2005

## USFKAD: An Expert System For Partial Differential Equations

Sami M. Kadamani  
*University of South Florida*

Follow this and additional works at: <https://digitalcommons.usf.edu/etd>



Part of the [American Studies Commons](#)

---

### Scholar Commons Citation

Kadamani, Sami M., "USFKAD: An Expert System For Partial Differential Equations" (2005). *USF Tampa Graduate Theses and Dissertations*.  
<https://digitalcommons.usf.edu/etd/712>

This Dissertation is brought to you for free and open access by the USF Graduate Theses and Dissertations at Digital Commons @ University of South Florida. It has been accepted for inclusion in USF Tampa Graduate Theses and Dissertations by an authorized administrator of Digital Commons @ University of South Florida. For more information, please contact [digitalcommons@usf.edu](mailto:digitalcommons@usf.edu).

USFKAD: An Expert System For Partial Differential Equations

by

Sami M. Kadamani

A dissertation submitted in partial fulfillment  
of the requirements for the degree of  
Doctor of Philosophy  
Department of Electrical Engineering  
College of Engineering  
University of South Florida

Major Professor: Arthur David Snider, Ph.D.  
Kenneth A. Buckle, Ph.D.  
Stanley C. Kranc, Ph.D.  
Wilfrido A. Moreno, Ph.D.  
Mohamed Elhamdadi, Ph.D.

Date of Approval:  
March 28, 2005

Keywords: eigen function, analytic solutions, pde, separation of variables, and symbolic  
computing

© Copyright 2005, Sami M. Kadamani

## **DEDICATION**

This dissertation is dedicated to my wife Daid, and my two children Serene and Tareq.  
Thank you for your patience, I would not have been able to accomplish this goal without  
your support.

## **ACKNOWLEDGEMENTS**

My deepest thanks to my advisor Dr. David Arthur Snider, for his belief in me and for both his guidance and support. I would like to extend my gratitude to Chairperson Sam Sakmar, Ph.D. and the members of the examining committee: Kenneth A. Buckle, Ph.D., Stanley C. Kranc, Ph.D., Wilfrido A. Moreno, Ph.D., and Mohamed Elhamdadi, Ph.D. In addition I would like to extend my thanks to my friends Husam Elrabi and Krista Morehead, whose input was invaluable.

## TABLE OF CONTENTS

LIST OF TABLES	iii
LIST OF FIGURES	iv
ABSTRACT	v
CHAPTER 1: INTRODUCTION	1
Need	1
Feasibility	3
CHAPTER 2: EXISTING SOFTWARE	5
CHAPTER 3: THE SEPARATION OF VARIABLES PROCEDURE AND ITS DECISION TREE STRUCTURE	8
CHAPTER 4: DISCUSSION AND FLOW CHART FOR USFKAD	14
Decision Tree	15
Fundamental Concept	19
Example 1	22
On-Screen Inquires	26
CHAPTER 5: APPLICATION EXAMPLES	28
Example 1	28
Example 2	30
Example 3	31
Example 4	33
Example 5	34
Example 6	35
CHAPTER 6: RECOMMENDATIONS FOR FUTURE DEVELOPMENTS	37
REFERENCES	38
APPENDICES	39
APPENDIX A: COMPARISON OF SOLVING PARTIAL DIFFERENTIAL EQUATIONS USING THE TRADITIONAL METHOD AND USFKAD	40

Solution 1	43
Solution 2	47
APPENDIX B: READ ME	50
APPENDIX C: TO USE USFKAD.PDF	53
ABOUT THE AUTHOR	End Page

## LIST OF TABLES

Table 1: PDE Equation Type Matrix	21
Table 2: PDE BC Matrix	21
Table 3: Matrix Example	23

## LIST OF FIGURES

Figure 1: USFKAD Flow Chart	16-20
Figure 2: PDE Problem	40
Figures 3: (a-d) Decomposition of the PDE Problem	40



# **USFKAD: AN EXPERT SYSTEM FOR PARTIAL DIFFERENTIAL EQUATIONS**

Sami M. Kadamani

## **ABSTRACT**

USFKAD is an encoded expert system for the eigenfunction expansion of solutions to the wave, diffusion, and Laplace equations: both homogeneous and nonhomogeneous; one, two, or three dimensions; Cartesian, cylindrical, or spherical coordinates; Dirichlet, Neumann, Robin, or singular boundary conditions; in time, frequency, or Laplace domain. The user follows a menu to enter his/her choices and the output is a LaTeX file containing the formula for the solution together with the transcendental equation for the eigenvalues (if necessary) and the projection formulas for the coefficients. The file is suitable for insertion into a book or journal article, and as a teaching aid. Virtually all cases are covered, including the Mellin, spherical harmonic, Bessel, modified Bessel, spherical Bessel, Dini, Hankel, Weber, MacDonald, and Kantorovich-Lebedev expansions, mixed spectrum, and rigid body modes.

# CHAPTER 1

## INTRODUCTION

### Need

Every practicing engineer whose specialty involves modeling of physical phenomena, such as electromagnetic fields, temperature, sound, stress and strain, fluid flow, diffusion, etc., has to deal with the mathematical syntax of the discipline - the partial differential equation (PDE). For example, the electrical engineering undergraduate classes in electromagnetic, semiconductor processing, thermal issues in electronic packaging, etc. should be able to call on this mathematical concept, at least peripherally, to provide the students some familiarity with the technical issues involved in the quantitative models. However, this subject (PDEs) is vast, complicated, and compromises have to be made in incorporating it into the undergraduate's curriculum. A 2-semester course that deals honestly and rigorously with the subject is out of the question.

The compromises presently employed at undergraduate institutions are:

- (1) A short treatment of PDEs that relies completely on numerical solvers; or
- (2) A brief tutorial that covers the basics of the separation of variables technique.

Each of these is unsatisfactory. (1) is inferior to (2) because, even with the graphic capabilities of today's hardware and software, it is extremely difficult for an inexperienced undergraduate user to tell, from a vast assemblage of tabulations and graphs, how the solutions will respond to changes in the boundary conditions or the

physical dimensions - issues of prime importance to engineering. For example consider the frequency of the resonant mode of a rectangular cavity with sides X, Y and Z given by:  $\omega = c\pi \sqrt{1/X^2 + 1/Y^2 + 1/Z^2}$  . These are not a terribly complicated formula, but contemplate trying to deduce it from graphs!

The eigenfunction expansions yielded by (2) do reveal these dependencies (and are exact). The drawback of this solution procedure is the lack of time to impart expertise in its implementation except for a few elementary cases - rectangular geometries and ideal conductors, for instance. The electrical world of cables, motors, and antennas is replete with cylindrical and spherical devices made of lousy materials, whose analyses entail Bessel functions and transcendental eigenvalue equations. The present-day curriculum has no room for the mastery of the "special functions" that occupied the toolbox of the 1950s engineer.

On the other hand, usually it is well within the capability of senior undergraduates to *verify* most features of an eigenfunction solution expansion.

Therefore an expert system, USFKAD, for partial differential equations (a smart software tool) that can automatically cull, from a library of eigenfunctions, the assemblage constituting the solutions to explicit problems, together with relevant graphics, would be a powerful enabler for undergraduate engineering training:

1. It would allow engineering analysis/design to proceed efficiently without being sidetracked by concerns of mathematical solvability.
2. As such, it would cut across many engineering disciplines.

3. It could be used to give a perspective on the separation-of-variables technique itself, by enabling “reverse-engineering” of the explicit solution formulas.  
(This will be elaborated below.)
4. In fact it would be a research tool that the engineer could continue to use in his professional career. Eigenfunction expansions are integral to the *mode-matching* procedure that is used in contemporary computational electromagnetism. And indeed, virtually every technical paper describing a new numerical solver compares its results with eigenfunction expansions, as testament to its accuracy.

### **Feasibility**

USFKAD, the subject of this dissertation, focuses on the theme that the mathematical structure, afforded by superposition, of the eigenfunction method for solving the separable PDEs of engineering can be expressed by a compact, universal, inviolate, and reasonably lucid algorithm; its formidability lies only in the details of its implementation - that is, in the enormous variety of eigenfunctions that must be employed for the curvilinear geometries. Thus it becomes feasible to contemplate a smart computer program that exploits this structure to judiciously select, from a library of eigenfunctions, the assemblage constituting the solutions to problems with explicit initial/boundary conditions.

In the subsequent chapters, more discussion and a list of examples of PDE solutions will be presented that progressively demonstrate the decision-tree nature of the general separation of variables procedure. This will exemplify the thesis that by reverse-

engineering explicit solution formulas one can experience a tutorial intercourse with the procedure itself.

## CHAPTER 2

### EXISTING SOFTWARE

Software for obtaining (analytic) solutions to *ordinary* differential equations exists in several forms, including Mathematica [6] and MAPLE [7]. It has not received universal adoption because extensive training in ordinary differential equations is already part of the required curriculum for all SMET (Science, Mathematics, Engineering, and Technology) students. For *partial* differential equations, MAPLE's *pdsolve* [Solution 1a] is a step in the right direction, but its arcane solution format provides little assistance for a non-expert in fitting the initial and boundary conditions that determine such dependencies. An example of its output, the electrostatic potential inside a sphere with charges distributed on the surface, is displayed as Solution 1a below. It is expressed (correctly) in terms of hyper-geometric and complex signum functions. But comparing this with the more recognizable solution display using USFKAD as shown in Solution 1b, one can clearly see the obvious simplification and straight forwardness of USFKAD.

#### **Solution 1a: Output from pdsolve**

$$\begin{aligned}
F(r, t, p) = & \text{}_{-C1} r^{(1/2\sqrt{1+4_{-c1}})} \text{}_{-C3} (-2 \sin(t)^2)^{(1/2\sqrt{-c_2})} \text{hypergeom}\left( \right. \\
& \left. \left[ \frac{1}{4} + \frac{1}{2}\sqrt{-c_2} + \frac{1}{4}\sqrt{1+4_{-c1}}, -\frac{1}{4}\sqrt{1+4_{-c1}} + \frac{1}{4} + \frac{1}{2}\sqrt{-c_2} \right], \left[ \frac{1}{2} \right], \frac{1}{2} \cos(2t) + \frac{1}{2} \right) \text{}_{-C5} \\
& \sin(\sqrt{-c_2} p) / \sqrt{r} + \text{}_{-C1} r^{(1/2\sqrt{1+4_{-c1}})} \text{}_{-C3} (-2 \sin(t)^2)^{(1/2\sqrt{-c_2})} \text{hypergeom}\left( \right. \\
& \left. \left[ \frac{1}{4} + \frac{1}{2}\sqrt{-c_2} + \frac{1}{4}\sqrt{1+4_{-c1}}, -\frac{1}{4}\sqrt{1+4_{-c1}} + \frac{1}{4} + \frac{1}{2}\sqrt{-c_2} \right], \left[ \frac{1}{2} \right], \frac{1}{2} \cos(2t) + \frac{1}{2} \right) \text{}_{-C6} \\
& \cos(\sqrt{-c_2} p) / \sqrt{r} + 2 \text{}_{-C1} r^{(1/2\sqrt{1+4_{-c1}})} \text{}_{-C4} (-2 \sin(t)^2)^{(1/2\sqrt{-c_2})} \text{csgn}(\cos(t)) \\
& \cos(t) \text{hypergeom}\left( \left[ \frac{3}{4} + \frac{1}{2}\sqrt{-c_2} + \frac{1}{4}\sqrt{1+4_{-c1}}, -\frac{1}{4}\sqrt{1+4_{-c1}} + \frac{3}{4} + \frac{1}{2}\sqrt{-c_2} \right], \left[ \frac{3}{2} \right], \right. \\
& \left. \frac{1}{2} \cos(2t) + \frac{1}{2} \right) \text{}_{-C5} \sin(\sqrt{-c_2} p) / \sqrt{r} + 2 \text{}_{-C1} r^{(1/2\sqrt{1+4_{-c1}})} \text{}_{-C4} \\
& (-2 \sin(t)^2)^{(1/2\sqrt{-c_2})} \text{csgn}(\cos(t)) \cos(t) \text{hypergeom}\left( \right. \\
& \left. \left[ \frac{3}{4} + \frac{1}{2}\sqrt{-c_2} + \frac{1}{4}\sqrt{1+4_{-c1}}, -\frac{1}{4}\sqrt{1+4_{-c1}} + \frac{3}{4} + \frac{1}{2}\sqrt{-c_2} \right], \left[ \frac{3}{2} \right], \frac{1}{2} \cos(2t) + \frac{1}{2} \right) \text{}_{-C6} \\
& \cos(\sqrt{-c_2} p) / \sqrt{r} + \text{}_{-C2} r^{(-1/2\sqrt{1+4_{-c1}})} \text{}_{-C3} (-2 \sin(t)^2)^{(1/2\sqrt{-c_2})} \text{hypergeom}\left( \right. \\
& \left. \left[ \frac{1}{4} + \frac{1}{2}\sqrt{-c_2} + \frac{1}{4}\sqrt{1+4_{-c1}}, -\frac{1}{4}\sqrt{1+4_{-c1}} + \frac{1}{4} + \frac{1}{2}\sqrt{-c_2} \right], \left[ \frac{1}{2} \right], \frac{1}{2} \cos(2t) + \frac{1}{2} \right) \text{}_{-C5} \\
& \sin(\sqrt{-c_2} p) / \sqrt{r} + \text{}_{-C2} r^{(-1/2\sqrt{1+4_{-c1}})} \text{}_{-C3} (-2 \sin(t)^2)^{(1/2\sqrt{-c_2})} \text{hypergeom}\left( \right. \\
& \left. \left[ \frac{1}{4} + \frac{1}{2}\sqrt{-c_2} + \frac{1}{4}\sqrt{1+4_{-c1}}, -\frac{1}{4}\sqrt{1+4_{-c1}} + \frac{1}{4} + \frac{1}{2}\sqrt{-c_2} \right], \left[ \frac{1}{2} \right], \frac{1}{2} \cos(2t) + \frac{1}{2} \right) \text{}_{-C6} \\
& \cos(\sqrt{-c_2} p) / \sqrt{r} + 2 \text{}_{-C2} r^{(-1/2\sqrt{1+4_{-c1}})} \text{}_{-C4} (-2 \sin(t)^2)^{(1/2\sqrt{-c_2})} \text{csgn}(\cos(t)) \\
& \cos(t) \text{hypergeom}\left( \left[ \frac{3}{4} + \frac{1}{2}\sqrt{-c_2} + \frac{1}{4}\sqrt{1+4_{-c1}}, -\frac{1}{4}\sqrt{1+4_{-c1}} + \frac{3}{4} + \frac{1}{2}\sqrt{-c_2} \right], \left[ \frac{3}{2} \right], \right. \\
& \left. \frac{1}{2} \cos(2t) + \frac{1}{2} \right) \text{}_{-C5} \sin(\sqrt{-c_2} p) / \sqrt{r} + 2 \text{}_{-C2} r^{(-1/2\sqrt{1+4_{-c1}})} \text{}_{-C4} \\
& (-2 \sin(t)^2)^{(1/2\sqrt{-c_2})} \text{csgn}(\cos(t)) \cos(t) \text{hypergeom}\left( \right. \\
& \left. \left[ \frac{3}{4} + \frac{1}{2}\sqrt{-c_2} + \frac{1}{4}\sqrt{1+4_{-c1}}, -\frac{1}{4}\sqrt{1+4_{-c1}} + \frac{3}{4} + \frac{1}{2}\sqrt{-c_2} \right], \left[ \frac{3}{2} \right], \frac{1}{2} \cos(2t) + \frac{1}{2} \right) \text{}_{-C6} \\
& \cos(\sqrt{-c_2} p) / \sqrt{r}
\end{aligned}$$

### Solution 1b: Output from USFKAD

$$\Psi = \Psi_1$$

$$\Psi_1 = \sum_{\ell=0}^{\infty} \sum_{m=-\ell}^{\ell} Y_{\ell m}(\phi, \theta) r^{\ell} A_{\ell m}$$

With

$$A_{\ell m} = \int_0^{\pi} \sin \phi d\phi \int_0^{2\pi} d\theta Y_{\ell m}^*(\phi, \theta)$$

$$M_r f_{r=b}(\theta, \phi)$$

$$M_r = \frac{1}{b^{\ell}}$$



### CHAPTER 3

#### THE SEPARATION OF VARIABLE PROCEDURE AND ITS DECISION TREE STRUCTURE

The three most common partial differential equations encountered in engineering physics are:

$$\left. \begin{array}{l} 1 \quad \text{Poisson's Equation} \quad \nabla^2 \Phi = f \\ 2 \quad \text{Diffusion Equation} \quad \frac{\partial \Phi}{\partial t} = \nabla^2 \Phi + f \\ 3 \quad \text{Wave Equation} \quad \frac{\partial^2 \Phi}{\partial t^2} = \nabla^2 \Phi + f \end{array} \right\} \quad (i)$$

These equations are solvable by the separation of variables process in the common coordinate systems: Cartesian, cylindrical (polar), and spherical.

The Separation of Variables Technique is the most important analytical method in engineering analysis for solving partial differential equations. The present exposition borrows heavily from [1]; see also [4 and 5]. The successful execution of this procedure for a given boundary value problem can be lengthy and tedious, because it involves three different mathematical procedures:

1. the use of superposition to decompose a complicated problem into a set of simpler ones;
2. the separation of the partial differential equation into a set of ordinary differential equations in an appropriate coordinate system; and

3. the construction of eigenfunction expansions (which are generalizations of the Fourier series) which satisfy the boundary conditions.

We shall briefly describe the solution procedure in such a way as to elucidate the algorithmic (decision-tree) nature. Then we describe a new code, USFKAD, which implements the algorithm in C++, generating a LaTeX output file for the solutions. Appendix A solution 1 shows part of a straightforward and uncomplicated problem solved using the traditional method and appendix A solution 2 using USFKAD. It could be seen that how cumbersome, time consuming, tedious, and treacherous it is to solve a problem manually.

The requisites for guaranteed success for the separation of variables are:

- 1 All boundary/initial conditions are applied at edges (surfaces, “manifolds”) where one of the independent variables is constant.
- 2 All boundary/initial conditions are linear, taking the generic format

$$\alpha\phi + \beta \frac{\partial\phi}{\partial n} = g \quad (ii)$$

$n$  is the coordinate that is constant on the boundary in question. If  $\beta = 0$  (ii) is called a Dirchlet condition (heat sink, electrical ground); if  $\alpha = 0$  (ii) is called a Neumann condition (ideal insulation, magnetic wall); otherwise it is a Robin condition (imperfect insulation). Also singular boundary conditions (singular points of the differential equation, boundaries at infinity, Sommerfeld radiation conditions, etc.) may be accommodated.

The given functions  $f$  or  $g$  (*i*), (*ii*) are called non-homogeneities; any equation is homogenous when  $f$  or  $g$  is zero. By linearity, any system can be decomposed into a superposition of systems in each of which only one equation is nonhomogenous.

Solutions to the homogenous forms of each of the three partial differential equations can be found in the form of products of functions depending on one independent variable only, in the appropriate coordinate systems [1]. In a wide range of engineering situations these one-dimensional functions are solutions of one of the following four types of ordinary differential equations:

- 1 Constant co-efficient (Cartesian coordinates  $x, y, z$ , time  $t$ , angle  $\theta$ )
- 2 Equidimensional (Cauchy-Euler) (polar coordinate radius)
- 3 Bessel (Cylindrical or spherical coordinate radius)
- 4 Legendre (azimuthal angle  $\phi$ )

Associated with the ordinary differential equation for each independent variable (other than time) are boundary conditions. After the decomposition step all of these boundary conditions are homogenous with one exception. For the unexceptional ordinary differential equations, the ordinary differential equation is second order and contains an unspecified parameter called the separation constant. The ordinary differential equation, separation constant, and boundary conditions constitute a Sturm-Louisville eigenvalue problem. The steps for solving such a problem are [1]:

1. With the differential equation expressed in the generic form

$$a_2(x)y'' + a_1(x)y' + a_0(x)y = \lambda g(x)y \quad (1)$$

write down the general solution as the sum of two independent particular solutions with undetermined coefficients:

$$y = C_1 y_1(x; \lambda) + C_2 y_2(x; \lambda). \quad (2)$$

The constant  $\lambda$  will appear as a parameter in the formulas.

Satisfy one of the boundary conditions:

$$\alpha y(a) + \beta y'(a) = 0, \gamma y(b) + \delta y'(b) = 0 \quad (3)$$

by the choice of  $C_1$  and  $C_2$ . In other words, use the relation

$$\alpha[C_1 y_1(a; \lambda) + C_2 y_2(a; \lambda)] + \beta[C_1 y_1'(a; \lambda) + C_2 y_2'(a; \lambda)] = 0 \quad (4)$$

to express  $C_1$  in terms of  $C_2$  or vice versa. The resulting solution is a constant multiple of

$$y(x; \lambda) = [\alpha y_2(a; \lambda) + \beta y_2'(a; \lambda)] y_1(x; \lambda) - [\alpha y_1(a; \lambda) + \beta y_1'(a; \lambda)] y_2(x; \lambda)$$

3. The remaining boundary condition

$$\gamma y(b; \lambda) + \delta y'(b; \lambda) = 0 \quad (6)$$

Is regarded as an equation for  $\lambda$ . Insert (5) and solve it for the eigenvalues  $\{\lambda_n\}$ , which will form a sequence going to  $\pm\infty$  according to whether  $a_2$  and  $g$  have opposite or the same signs. The  $n^{\text{th}}$  eigenfunction  $\phi_n(x) = y(x; \lambda_n)$  should have  $n-1$  interior zeros (assuming that the eigen values are enumerated from  $n=1$ ).

For the Dirichlet condition as an example of step (2): with differential equations  $\Psi'' = \lambda\Psi$ , and boundary conditions:  $\Psi(0) = 0$  and  $\Psi'(X) = 0$  the general solution is:  $\Psi = C_1 \cos\sqrt{\lambda}X + C_2 \sin\sqrt{\lambda}X$ . Applying the boundary condition  $\Psi(0) = 0, C_1(1) + C_2(0) = 0$ , yields  $\Psi = C \sin\sqrt{\lambda}X$ .

For the Robin condition as another example: with differential equation  $\Psi'' = \lambda\Psi$ , and boundary conditions:  $\Psi'(0) + \alpha_{x=0}\Psi(0) = 0$  and  $\Psi'(X) + \alpha_{x=X}\Psi(X) = 0$  the general solution is  $\Psi = C_1 \cos\sqrt{\lambda}X + C_2 \sin\sqrt{\lambda}X$ ; applying the boundary condition  $y'(0) + \alpha_{x=0}\Psi(0) = -C_1\lambda(0) + C_2\sqrt{\lambda}(1) + \alpha_{x=0}[C_1(1) + C_2(0)] = C_2\sqrt{\lambda} + C_1\alpha_{x=0}$  yields  $\Psi = C[\lambda \cos\sqrt{\lambda}X - \alpha_{x=0} \sin\sqrt{\lambda}X]$ .

The ordinary differential equation in the final independent variable has a non-homogenous boundary/initial condition and no unspecified parameters; it is not an eigenvalue problem, but its “basic” solution, satisfying the homogenous boundary condition only, is found as above.

The general separation of variables solution for a three-dimensional problem with a single boundary non-homogeneity looks like:

1. Coordinates  $\xi, \eta, \nu$  {Cartesian, cylindrical, or spherical}

$$\Sigma_{\alpha,\beta,\delta} A_{\alpha,\beta,\delta} \Xi_{\alpha}(\xi) H_{\beta}(\eta) N_{\delta}(\nu) T_{\alpha,\beta,\delta}(t)$$

2. T is a sinusoid in time (wave equation) or exponential (diffusion equation) or not present (time-independent Poisson’s equation, or frequency or Laplace domain).

3. Two of the factors  $\Xi$ ,  $H$ ,  $N$  each satisfy a second order ordinary differential equation with a parameter and one homogenous boundary condition (holding for all values of the parameter), while the specified values for the parameter enforce the second boundary condition. The remaining factor satisfies only one homogenous boundary condition, and its parameter is fixed by the others (through the partial differential equation).
4. The coefficients  $A_{\alpha,\beta,\delta}$  are determined from the boundary/initial conditions by orthogonality.

## CHAPTER 4

### DISCUSSION AND FLOW CHART FOR USFKAD

The coding of the expert system USFKAD was completed and successfully tested. The C++ [8] source contains about 5,000 lines of code and occupies 211 KB of computer memory. The executable file occupies 1300 KB. USFKAD utilizes the Separation of Variables technique.

USFKAD has been encoded as an expert system for the eigenfunction expansion of solutions to the wave, diffusion, and Laplace equations: both homogeneous and nonhomogeneous; one, two, or three dimensions; Cartesian, cylindrical, or spherical coordinate systems; Dirichlet, Neumann, Robin, or singular boundary conditions; in time, frequency, or Laplace domain. The user follows a menu to enter his/her choices and the output is a LaTeX file containing the formula for the solution together with the transcendental equation for the eigenvalues (if necessary) and the orthogonal-projection formulas for the coefficients. The output file is suitable, among other things, for insertion into a book or journal article, and as a teaching aid. Virtually all cases are covered, including the Mellin, spherical harmonic, Bessel, modified Bessel, spherical Bessel, Dini, Hankel, Weber, MacDonald, and Kantorovich-Lebedev expansions, mixed spectrum, and rigid body modes.

The enabling attribute of this expert system is the observation that a decision-tree algorithm can be constructed to assemble the eigenfunctions needed for any particular

problem. Input to the algorithm would be the particular PDE, the solution domain (time, Laplace, or frequency), and the geometry and boundary conditions. The software begins by scanning the boundaries looking for a nonhomogeneous boundary condition. It then assembles the (non-eigenfunction) factor for this direction and the eigenfunction factors for the other directions, writes this to a file, and moves on to the next boundary. Finally it assembles and writes the Green's function terms if the PDE is nonhomogeneous, and the transient or oscillatory terms. By passing generic variable names between subroutines and exploiting the similarity of the logic for the Laplace, frequency, and time-independent cases, the program needs only about 200 eigenfunction subroutines to cover all the possibilities. Each such subroutine contains the requisite normalization constants and the transcendental equations defining the eigenvalues.

### **Decision Tree**

The flow chart below (Figure 1) depicts and describes the logic behind USFKAD. Following the chart, one can be guided to reach a successful, complete, and comprehensive solution to any appropriate partial differential equation along with its specific boundary conditions. This flow chart represents the brain of USFKAD.



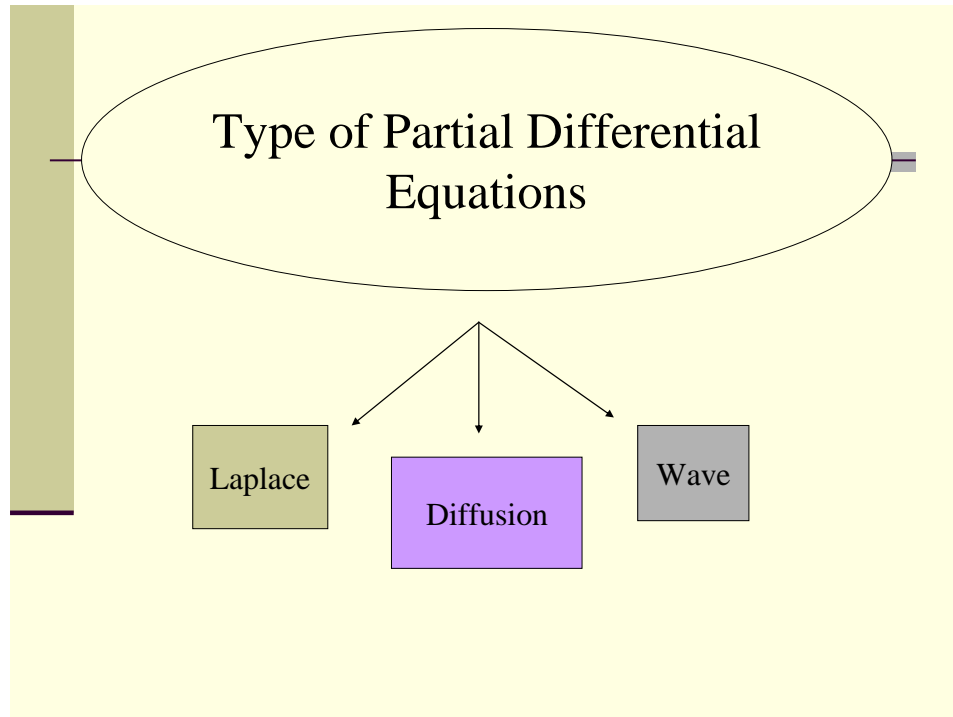


Figure 1: USFKAD Flow Chart

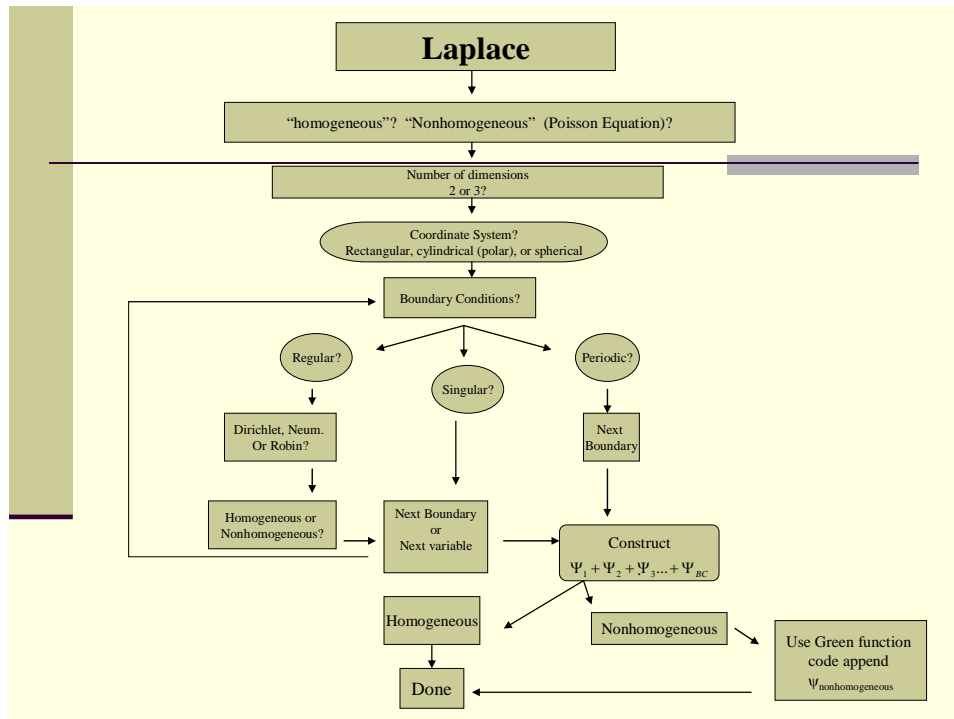


Figure 1: Continued

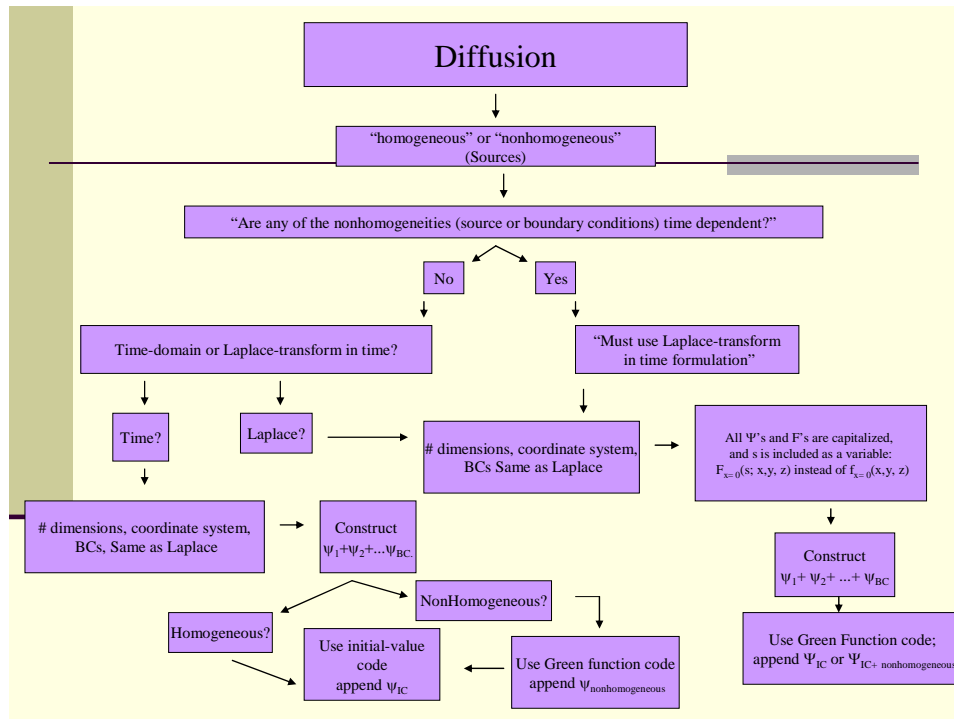


Figure 1: Continued

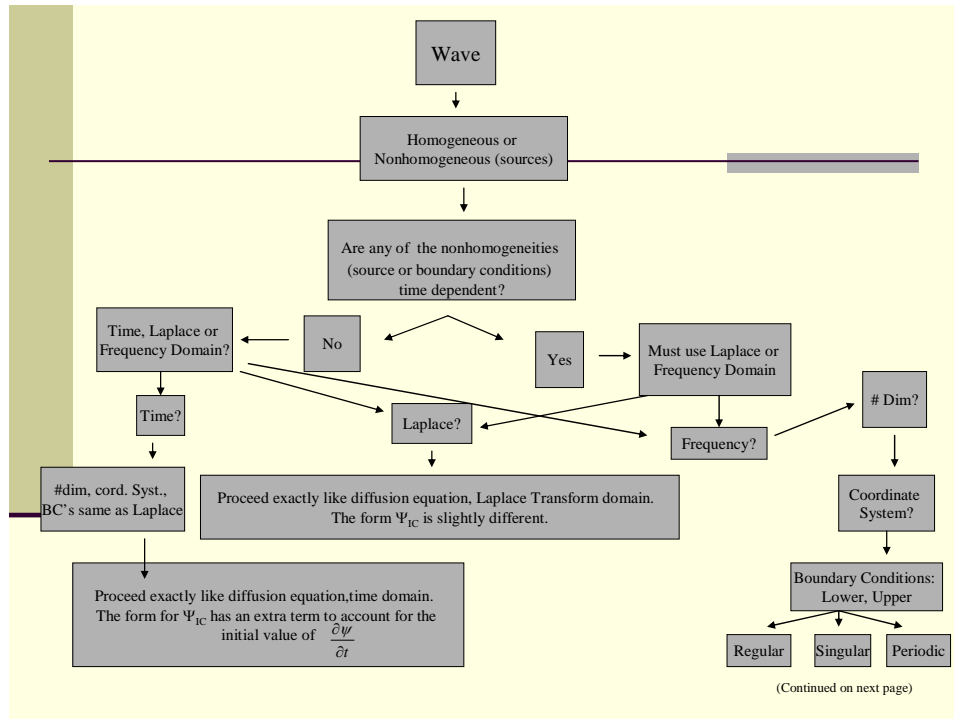


Figure 1: Continued

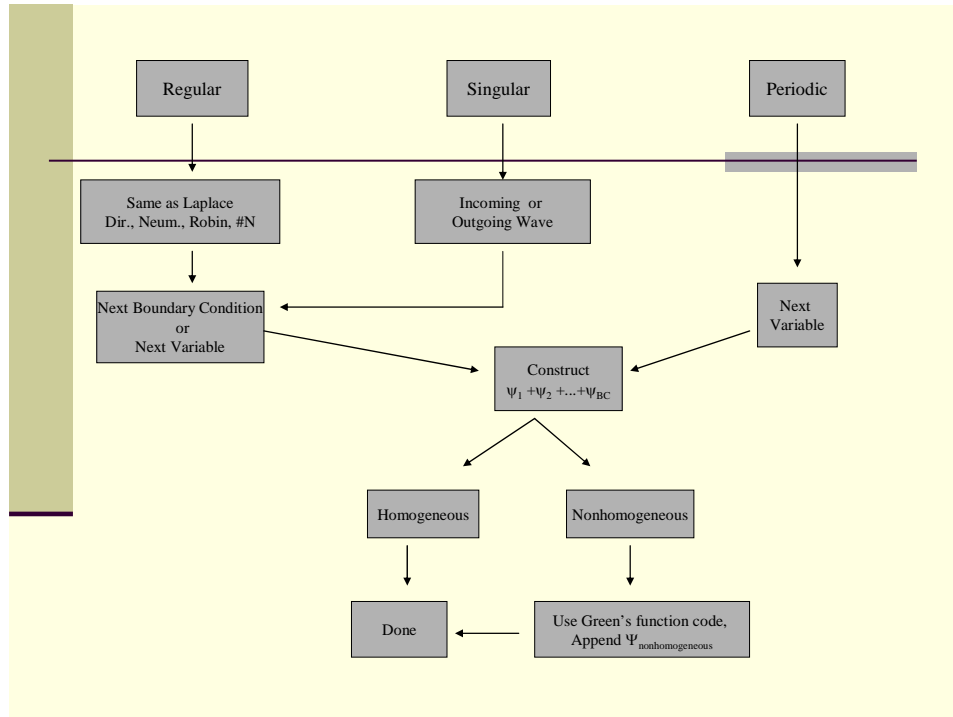


Figure 1: Continued

## Fundamental Concept

The fundamental concept of USFKAD is contained in the following two matrices: the PDE type matrix (table 1) and the PDE boundary condition (BC) matrix (table 2) as illustrated below.

**Table 1: PDE Equation Type Matrix**

PDE Equation Type Matrix							
Zero	One	Two	Three	Four	Five	Six	Seven
eqtype							
PDE	Dimension	Coordinate System	Coordinate 1	Coordinate 2	Coordinate 3	Time	Homogeneous or Nonhomogeneous?
0=Laplace	1,2,3	0=rectangular	x=0;	y=1	z=2	t=8	H=0, N=1
1=Diffusion, t		1=polar/ cylindrical	theta=3;	r2d=4			
2=Diffusion, s		2=spherical	phi=8	theta3=9	r3D=6		
3=Wave, t			theta=3;	z=2	Rho=5		
4=Wave, s							
5=Wave, omega							

**Table 2: PDE BC Matrix**

PDE Boundary Condition Matrix										
	Zero	One	Two	Three	Four	Five	Six	Seven	Eight	Nine
Row #0	R or S	D,N,R	N,H	CC	R or S	D,N,R,O,I	N,H	<b>x</b>	0 (zero)	X
Row #1	R or S	D,N,R	N,H	CC	R or S	D,N,R,O,I	N,H	<b>y</b>	0 (zero)	Y
Row #2	R or S	D,N,R	N,H	CC	R or S	D,N,R,O,I	N,H	<b>z</b>	0 (zero)	Z
Row #3	R	D,N,R,P	N,H	CC	R	D,N,R,P	N,H	<b>theta</b>	0 (zero)	THETA
Row #4	R or S	D,N,R	N,H	CE	R or S	D,N,R,O,I	N,H	<b>r (2d)</b>	a	b
Row #5	R or S	D,N,R	N,H	BB	R or S	D,N,R,O,I	N,H	<b>rho</b>	a	b
Row #6	R or S	D,N,R	N,H	SB	R or S	D,N,R,O,I	N,H	<b>r3D</b>	a	b
Row #7	R	D	N,H	CC	R	D	N,H	<b>t</b>	0 (zero)	0 (zero)
Row #8	S	S	H	LG	S	S	H	<b>phi</b>	0 (zero)	pi
Row #9	R or S	P	H	SH	R or S	P	H	<b>theta3d</b>	0 (zero)	2 pi

The PDE type matrix (table 1) identifies all types of PDE's including the number of dimensions, coordinate system, coordinates, whether it is time dependent or not, and whether it is homogeneous or not.

Now once the PDE Equation type has been selected the Boundary conditions must be identified. The BC matrix virtually accounts for all types of PDE's. Using the PDE type matrix and the BC matrix, we came up with 200 different possible PDE cases or scenarios that virtually cover and provide solutions for all PDE problems. To accomplish this there are 200 subroutines in USFKAD code which provides solutions for each possible PDE problems.

### **Example 1**

Consider Laplace's equation in a box, Homogeneous Dirichlet conditions at  $x=0$  and  $X$ , homogeneous Dirichlet at  $y=0$ , nonhomogeneous Dirichlet at  $y=Y$ . homogeneous Neumann condition at  $z=0$ , nonhomogeneous Neumann at  $z=Z$ . As the user inputs these data, USFKAD fills in the BC matrix as shown on table 3.

**Table 3: Matrix Example**

	Column #0	#1	#2	#3	#4	#5	#6	#7	#8	#9
	eqtype									
	PDE	dim	coord	coord 1	coord 2	coord 3	time	H, N?		
	0=Laplace	3	0=rect	x=0;	y=1	z=2		H=0		
	Boundary	Conditions	"mtrx"							
Row #0	Regular	D	H	CC	R	D	H	x	0 (zero)	X
Row #1	Regular	D	H	CC	R	D	N	y	0 (zero)	Y
Row #2	Regular	N	H	CC	R	N	N	z	0 (zero)	Z
Row #3										
Row #4										
Row #5										
Row #6										
Row #7										
Row #8										
Row #9										

Note that final solution will be

$$\sum_{j=1}^{\infty} \sum_{k=0}^{\infty} a_{jk} \sin \frac{j\pi x}{X} \cos \frac{k\pi z}{Z} \sinh \sqrt{\left(\frac{j\pi}{X}\right)^2 + \left(\frac{k\pi}{Z}\right)^2} y$$

$$+ \sum_{j=1}^{\infty} \sum_{k=1}^{\infty} b_{jk} \sin \frac{j\pi x}{X} \sin \frac{k\pi y}{Y} \cosh \sqrt{\left(\frac{j\pi}{X}\right)^2 + \left(\frac{k\pi}{Y}\right)^2} z$$

with

$$a_{jk} = \frac{1}{\int_0^X \sin^2 \frac{j\pi x}{X} dx} \frac{1}{\int_0^Z \cos^2 \frac{k\pi z}{Z} dz} \int_0^X \int_0^Z f_{y=y}(x, z) \sin \frac{j\pi x}{X} \cos \frac{k\pi z}{Z} dz dx.$$



$$b_{jk} = \frac{1}{\int_0^X \sin^2 \frac{j\pi x}{X} dx} \frac{1}{\int_0^Y \sin^2 \frac{k\pi y}{Y} dy} \int_0^X \int_0^Y f_{z=z}(x, y) \sin \frac{j\pi x}{X} \sin \frac{k\pi y}{Y} dy dx.$$

USFKAD scans down column 2, and then column 6, until it sees an N (nonhomogeneous) in row 1, column 6. It begins to assemble the solution to the sub problem where all BCs are homogeneous except at  $y=Y$ . Saving  $y$  for last, it looks at the  $x$ -row, row #0. It sees the specifications

1. Constant coefficient DE (from column 3),
2. Dirichlet BC at the low end (from column 1),
3. Dirichlet BC at the high end (from column 5),
4. Homogeneous BC at both ends (because  $y$  has the only nonhomogeneous).

So USFKAD assembles the word "CCDDHH" from these columns and sends this word and the symbol "x" to the eigenfunction searcher. The latter returns the eigenfunction as "sin", the eigenvalues as " $\kappa_x = \pi/X, 2\pi/X, \dots$ ", and the superposition type as " $\Sigma_{\kappa_x}$ ". (If the eigenvalues formed a continuum, the superposition type would be

" $\int d\kappa_x$ "). It also returns the formula for  $\frac{1}{\int_0^X \sin^2 \frac{j\pi x}{X} dx}$  and the weight factor for this DE

("1", hence blank, in this case.)

USFKAD writes each of these returned formulas to separate lists, and looks for the next variable,  $z$ . It reads off the characteristics from row #2 of the BC matrix and sends " $z, CCNNHH$ " to the eigenfunction sorter, which returns the appropriate formulas as before. They are concatenated with the earlier formulas in the corresponding lists.

Then USFKAD fetches the symbols for the y factor by sending "y, CCDDHN,  $\kappa_x$ ,  $\kappa_z$ ", as read from row #1 and the previous lists, to the eigenfunction sorter (of course this factor is not literally an eigenfunction). The latter returns the sinh function and the  $\sqrt{\kappa_x^2 + \kappa_z^2}$  symbol, and USFKAD concatenates these with the other lists and merges all the lists together, calling the result  $\Psi_1$ .

Next USFKAD continues to scan columns 2 and 6, looking for the next "N" (nonhomogeneous BC) marker. It finds "N" in row #2, column #6, so it assembles x, y eigenfunctions and z-factors as before.

If the PDE itself is nonhomogeneous (as marked in column 7 of eqtype), USFKAD assembles a sum for the eigenfunction expansion of the Green's function ( $\Psi_3$ , example 1, chapter 5).

In the time domain (as marked in column 0 of eqtype), USFKAD assembles the time factors from the eigenvalues symbols it has accumulated (example 4, chapter 5).

In the frequency or Laplace domain (as marked in column 0 of eqtype), USFKAD sends the eigenvalues ( $\kappa_x, \dots$ ) and the transform variable (s or  $\omega$ ) (in accordance with column #0 of eqtype) to the subroutine searcher for the nonhomogeneous factor. These nonhomogeneous-factor subroutines simply incorporate them into the formula display ( $\sqrt{\kappa_x^2 + s}$  in example 5 and  $\sqrt{\omega^2 - \kappa_x^2 - \kappa_y^2}$  in example 6, chapter 5).

### **On-Screen Inquires**

USFKAD is invoked by answering a list of printed on-screen inquiries as follows:  
Select the Partial Differential Equation:

0 - Laplace or Poisson:  $\text{Laplacian } \Psi + f(\text{interior}) = 0$ ;

1 - Diffusion, Time Domain:  $d \Psi / dt = \text{Laplacian } \Psi + f(\text{interior})$ ;

2 - Diffusion, s-plane:  $s \Psi - \Psi(t=0) = \text{Laplacian } \Psi + F(\text{interior})$ ;

3 - Wave, Time Domain:  $d^2 \Psi / dt^2 = \text{Laplacian } \Psi + f(\text{interior})$ ;

4 - Wave, s-plane:  $s^2 \Psi - s\Psi(t=0) - \Psi'(t=0) = \text{Laplacian } \Psi + F(\text{interior})$ ;

5 - Wave, frequency domain:  $-\omega^2 \Psi = \text{Laplacian } \Psi + F(\text{interior})$ .

Is the PDE homogeneous (enter 0) or nonhomogeneous (enter 1)?

Enter 1, 2, or 3 for 1, 2, or 3 dimensions.

Select the Coordinate System: 0 - Rectangular, 1 - Cylindrical or Polar, 2 - Spherical.

Select the boundary condition at the lower (upper) end for the coordinate  $x$  ( $y$ ,  $z$ ,  $r$ ,  $\rho$ , and  $\theta$ ):

Enter 1 for Dirichlet, Homogeneous;

Enter 2 for Dirichlet, Nonhomogeneous;

Enter 3 for Neumann, Homogeneous;

Enter 4 for Neumann, Nonhomogeneous;

Enter 5 for Robin, Homogeneous;

Enter 6 for Robin, Nonhomogeneous;

Enter 7 for Periodic Boundary Conditions;

Enter 8 for Singular Boundary Condition;

Enter 9 for Singular, Sommerfeld Outgoing Wave Condition;

Enter 10 for Singular, Sommerfeld Incoming Wave Condition.

The output of USFKAD is a LaTeX file. It requires subsequent processing by TeX software, which must be resident on the user's computer.

The above procedure and methodology are further illustrated through the six different examples listed in the next chapter.

Further details on the lexicon of the software in expressing the physical dimensions and boundary conditions appear in Appendices B and C, respectively.

## CHAPTER 5

### APPLICATION EXAMPLES

This chapter contains a list of examples of partial differential equation solutions that progressively demonstrate the decision tree nature of the general separation of variable procedure. Physical interpretations of these mathematical problems are stated here in terms of heat and sound phenomena; they all have electrical counterparts.

#### Example 1

Steady state heat flow in a rectangle with edge and interior heat sources

(nonhomogeneous Laplace/Poisson equation in two dimensions, rectangular coordinates,

Dirichlet conditions on two sides, Neumann conditions on two sides):

$$\begin{aligned}\nabla^2\Psi &= -f_{\text{interior}}(x, y) \\ \Psi(0, y) &= 0, \Psi(X, y) = f_{x=X}(y) \\ \frac{\partial\Psi}{\partial y}(x, 0) &= f_{y=0}(x), \frac{\partial\Psi}{\partial y}(x, Y) = 0\end{aligned}$$

The solution is as follows:

$$\Psi = \Psi_1 + \Psi_2 + \Psi_3$$

$$\Psi_1 = \sum_{\kappa_x} \cos\kappa_x x \eta_y(y; \kappa_x) A(\kappa_x)$$

with

$$\kappa_x = 0, \frac{\pi}{X}, \frac{2\pi}{X}, \frac{3\pi}{X}, \dots$$

$$\eta_y(y; \kappa_x) = \begin{cases} Y-y & \text{if } \kappa_x=0; \\ \sinh \kappa_x (Y-y) & \text{otherwise.} \end{cases}$$

$$A(\kappa_x) = \int_0^X dx \cos \kappa_x x N_{\kappa_x} M_{\kappa_x} f_{y=0}(x)$$

$$N_{\kappa_x} = \begin{cases} \frac{1}{X} & \text{if } \kappa_x=0; \\ \frac{2}{X} & \text{otherwise} \end{cases}$$

$$M_{\kappa_x} = \begin{cases} \frac{1}{Y} & \text{if } \kappa_x=0; \\ \frac{1}{\sinh \kappa_x Y} & \text{otherwise.} \end{cases}$$

$$\Psi_2 = \sum_{\kappa_y} \sin \kappa_y y \cosh \kappa_y x A(\kappa_y)$$

with

$$\kappa_y = \frac{\pi}{Y}, \frac{2\pi}{Y}, \frac{3\pi}{Y}, \dots$$

$$A(\kappa_y) = \int_0^Y dy \sin \kappa_y y \frac{2}{Y} M_{\kappa_y} f_{x=X}(y)$$

$$M_{\kappa_y} = \begin{cases} 0 & \text{if } \kappa_y=0; \\ \frac{1}{\kappa_y \sinh \kappa_y X} & \text{otherwise.} \end{cases}$$

$$\Psi_3 = \sum_{\kappa_x} \sum_{\kappa_y} \cos \kappa_x x \sin \kappa_y y A(\kappa_x, \kappa_y)$$

with

$$\kappa_x = 0, \frac{\pi}{X}, \frac{2\pi}{X}, \frac{3\pi}{X}, \dots$$

$$\kappa_y = \frac{\pi}{Y}, \frac{2\pi}{Y}, \frac{3\pi}{Y}, \dots$$

$$A(\kappa_x, \kappa_y) = \int_0^X dx \int_0^Y dy \cos \kappa_x x N_{\kappa_x} \sin \kappa_y y \frac{2}{Y} \frac{f_{\text{interior}}(x, y)}{\kappa_x^2 + \kappa_y^2}$$

$$N_{\kappa_x} = \begin{cases} \frac{1}{X} & \text{if } \kappa_x=0; \\ \frac{2}{X} & \text{otherwise.} \end{cases}$$

Here we see some of the features of separation of variables:

1. The basic decomposition of the problem into three subproblems, each of which contains only one nonhomogeneous equation.
2. Each subsolution expressed as a sum of terms containing an eigenfunction factor, satisfying homogeneous boundary conditions at each end, and a “non-eigenfunction” factor, satisfying a homogeneous boundary condition at one end only (note the exceptional form of the latter factor in  $\Psi_2$ , and of the normalization constants for the cosine’s “DC” term when  $\kappa_x = 0$ );
3. Coefficients computed by orthogonality to give the corresponding nonhomogeneity;
4. The construction of the Green’s function out of the same eigenfunctions.

### Example 2

Steady state heat flow in a cube with facial heat sources and imperfect facial insulation (homogeneous Laplace equation in three dimensions, rectangular coordinates, homogeneous Dirichlet conditions on four sides, nonhomogeneous Dirichlet condition on one side, homogeneous Robin condition on one side):

$$\begin{aligned} \nabla^2 \Psi &= 0 \\ \Psi(0, y, z) &= 0, \Psi(X, y, z) = 0 \\ \frac{\partial \Psi}{\partial z}(x, y, 0) &= 0, \frac{\partial \Psi}{\partial z}(x, y, Z) = f_{z=Z}(x, y) \\ \Psi(x, 0, z) &= 0, \frac{\partial \Psi}{\partial y}(x, Y, z) + \alpha_{y=Y} \Psi(x, Y, z) = 0 \end{aligned}$$

The solution is

$$\Psi = \sum_{\kappa_x} \sum_{\kappa_y} \sin \kappa_x x \eta_y(y; \kappa_y) \cosh \sqrt{\kappa_x^2 + \kappa_y^2} z A(\kappa_x, \kappa_y)$$

with

$$\kappa_x = \frac{\pi}{X}, \frac{2\pi}{X}, \frac{3\pi}{X}, \dots$$

$\eta_y(y; \kappa_y) = \sin \kappa_y y$  where  $\{\kappa_y\}$  are the nonzero solutions (possibly imaginary) to

$$\alpha_y \sin \kappa_y Y + \cos \kappa_y Y = 0$$

also if  $\alpha_y Y + 1 = 0$  include  $\kappa_y = 0$ ,  $\eta_y(y; 0) = y$ .

$$A(\kappa_x, \kappa_y) = \int_0^X dx \int_0^Y dy \sin \kappa_x x \frac{2}{X} \eta_y(y; \kappa_y) N_{\kappa_y} M_{\sqrt{\kappa_x^2 + \kappa_y^2}} f_{z=Z}(x, y)$$

$$N_{\kappa_y} = \begin{cases} \frac{3}{Y^3} & \text{if } \kappa_y = 0; \\ \frac{2}{Y + (\alpha_y + 1)/(\alpha_y^2 + \kappa_y^2)} & \text{otherwise.} \end{cases}$$

$$M_{\sqrt{\kappa_x^2 + \kappa_y^2}} = \begin{cases} 0 & \text{if } \sqrt{\kappa_x^2 + \kappa_y^2} = 0; \\ \frac{1}{\sqrt{\kappa_x^2 + \kappa_y^2} \sinh \sqrt{\kappa_x^2 + \kappa_y^2} Z} & \text{otherwise.} \end{cases}$$

This example illustrates the straightforward extension of the procedure to three dimensions and the transcendental equation that the Robin boundary condition invokes for the eigenvalues.

### Example 3

Steady state heat flow in a cylindrical sector with facial heat sources (homogenous Laplace equation in the three dimensions inside a partial cylinder, nonhomogenous Dirichlet condition on the top and one flat side, homogenous Dirichlet conditions on the bottom and the curved wall, and a homogenous Neumann condition on the other flat side):



$$\begin{aligned}\nabla^2 \Psi &= 0 \\ \Psi(\rho, \theta, 0) &= \Psi(b, \theta, z) = 0, \\ \frac{\partial \Psi}{\partial \theta}(\rho, 0, z) &= 0, \quad \Psi(\rho, \Theta, z) = f_{\theta=\Theta}(z, \rho), \\ \Psi(\rho, \theta, Z) &= f_{z=Z}(\theta, \rho)\end{aligned}$$

The solution to this system is expressed

$$\begin{aligned}\Psi &= \Psi_1 + \Psi_2 \\ \Psi_1 &= \sum_{\kappa_z} \int_0^\infty d\kappa_{\rho:z} \sin \kappa_z z [K_{i\kappa_{\rho:z}}(\kappa_z b) I_{i\kappa_{\rho:z}}(\kappa_z \rho) - I_{i\kappa_{\rho:z}}(\kappa_z b) K_{i\kappa_{\rho:z}}(\kappa_z \rho)] \times \cosh K_{\rho:z} \theta A(\kappa_z, \kappa_{\rho:z})\end{aligned}$$

with

$$\begin{aligned}\kappa_z &= \frac{\pi}{Z}, \frac{2\pi}{Z}, \frac{3\pi}{Z}, \dots \\ A(\kappa_z, \kappa_{\rho:z}) &= \int_0^Z dz \int_0^b d\rho \sin \kappa_z z \frac{2}{Z} [K_{i\kappa_{\rho:z}}(\kappa_z b) I_{i\kappa_{\rho:z}}(\kappa_z \rho) - I_{i\kappa_{\rho:z}}(\kappa_z b) K_{i\kappa_{\rho:z}}(\kappa_z \rho)] \frac{1}{\rho} \left[ \frac{2\kappa_{\rho:z} \sinh \kappa_{\rho:z} \pi}{\pi^2 [I_{i\kappa_{\rho:z}}(\kappa_z b)]^2} \frac{1}{\cosh \kappa_{\rho:z} \Theta} f_{\theta=\Theta}(z, \rho) \right] \\ \Psi_2 &= \sum_{\kappa_\theta} \sum_{\kappa_{\rho:\theta}} \cos \kappa_\theta \theta J_{\kappa_\theta}(\kappa_{\rho:\theta} \rho) \eta_z(z; \kappa_{\rho:\theta}) A(\kappa_\theta, \kappa_{\rho:\theta})\end{aligned}$$

with

$$\kappa_\theta = \frac{\pi}{2\Theta}, \frac{3\pi}{2\Theta}, \frac{5\pi}{2\Theta}, \dots$$

For each value of  $\kappa_\theta, \kappa_{\rho:\theta} = \frac{j_{\kappa_\theta, \kappa_{\rho:\theta}}}{b}$  where  $\{j_{\kappa_\theta, \kappa_{\rho:\theta}}\}$  are the positive roots of  $J_{\kappa_\theta}(j_{\kappa_\theta, \kappa_{\rho:\theta}}) = 0$

$$\eta_z(z; \kappa_{\rho:\theta}) = \begin{cases} z & \text{if } \kappa_{\rho:\theta} = 0; \\ \sin \kappa_{\rho:\theta} z & \text{otherwise.} \end{cases}$$

$$A(\kappa_\theta, \kappa_{\rho:\theta}) = \int_0^\Theta d\theta \int_a^b d\rho \cos \kappa_\theta \theta \frac{2}{\Theta} J_{\kappa_\theta}(\kappa_{\rho:\theta} \rho) \rho \frac{2}{b^2 J_{\kappa_\theta+1}^2(j_{\kappa_\theta, \kappa_{\rho:\theta}})} M_{\kappa_{\rho:\theta}} f_{z=Z}(\theta, \rho)$$

$$M_{\kappa_{\rho:\theta}} = \begin{cases} \frac{1}{z} & \text{if } \kappa_{\rho:\theta} = 0, \\ \frac{1}{\sin \kappa_{\rho:\theta} Z \theta} & \text{otherwise.} \end{cases}$$

This example illustrates the singular boundary condition for  $\rho = 0$  and the Lebedev eigen

functions in  $\rho$  included by the nonhomogenous condition for  $\theta = \Theta$ . The singular

boundary condition includes a continuous, rather than discrete, spectrum. Other than [1],

the Lebedev expansions do not appear in any English language mathematics textbook

except [3], where their correctness is betrayed by a persistent systematic error in the

tabulations. Their omission is probably due to their intimidating nomenclature (note the

analytic condition of the subscript into the complex plane). Ignoring them is, however, criminal, because (as we see) they occur in realistic problems; indeed, in the analysis of edge diffraction [2] they are crucial.

#### Example 4

Sound wave inside a sphere (homogenous wave equations inside a sphere, time-independent nonhomogenous Dirichlet conditions on the surface):

$$\frac{\partial^2 \Psi}{\partial t^2} = \nabla^2 \Psi$$

$$\Psi(r = b, \theta, \phi, t) = f_{r=b}(\theta, \phi)$$

The solution to this system is expressed

$$\Psi = \Psi_{steady\ state} + \Psi_{transient}$$

$$\Psi_{transient} = \Psi_{transient\ #1} + \Psi_{transient\ #2}$$

$$\Psi_{steady\ state} = \sum_{\ell=0}^{\infty} \sum_{m=-\ell}^{\ell} Y_{\ell m}(\phi, \theta) r^{\ell} A_{\ell m}$$

with

$$A_{\ell m} = \int_0^{\pi} \sin \phi d\phi \int_0^{2\pi} d\theta Y_{\ell m}^*(\phi, \theta) M_r f_{r=b}(\theta, \phi)$$

$$M_r = \frac{1}{b^{\ell}}$$

$$\Psi_{transient\ #1} = \sum_{\ell=0}^{\infty} \sum_{m=-\ell}^{\ell} \sum_{p=1}^{\infty} Y_{\ell m}(\phi, \theta) j_{\ell}(\kappa_r r) \cos \kappa_r t A_{\ell m}(\kappa_r)$$

with

$$\kappa_r = s_{\ell, p} / b \text{ where } s_{\ell, p} \text{ is the } p\text{th positive zero of } j_{\ell}.$$

$$A_{\ell m}(\kappa_r) = \int_0^{\pi} \sin \phi d\phi \int_0^{2\pi} d\theta \int_0^b r^2 dr Y_{\ell m}^*(\phi, \theta) j_{\ell}(\kappa_r r) \frac{2}{b^3 j_{\ell+1}^2(s_{\ell, p})} [\Psi(\theta, \phi, r; 0) - \Psi_{steady\ state}]$$

$$\Psi_{transient\ #2} = \sum_{\ell=0}^{\infty} \sum_{m=-\ell}^{\ell} \sum_{p=1}^{\infty} Y_{\ell m}(\phi, \theta) j_{\ell}(\kappa_r r) \frac{\sin \kappa_r t}{\kappa_r} A_{\ell m}(\kappa_r)$$

with

$\kappa_r = s_{\ell,p} / b$  where  $s_{\ell,p}$  is the pth positive zero of  $j_\ell$ .

$$A_m(\kappa_r) = \int_0^\pi \sin \phi d\phi \int_0^{2\pi} d\theta \int_0^b r^2 dr Y_{\ell m}^*(\phi, \theta) j_\ell(\kappa_r r) \frac{2}{b^3 j_{\ell+1}^2(s_{\ell,p})} \frac{\partial \Psi(\theta, \phi, r; 0)}{\partial t}$$

This example demonstrates the decomposition of the solution into a steady-state component and a transient component. The time functions can be treated just like the non-eigenfunctions in the previous solutions, except that they satisfy initial conditions instead of boundary conditions.

### Example 5

Transient heat flow in a rectangle with transient interior and edge heat sources (nonhomogenous diffusion equations, two dimensions, rectangular coordinates, homogenous Dirichlet conditions on two sides, homogenous Neumann condition on one side, time-dependent nonhomogenous Neumann conditions on one side):

$$\begin{aligned} \frac{\partial \Psi}{\partial t} &= \nabla^2 \Psi + f_{\text{interior}}(x, y, t) \\ \Psi(0, y) &= 0, \Psi(X, y) = 0 \\ \frac{\partial \Psi}{\partial y}(x, 0) &= 0, \frac{\partial \Psi}{\partial y}(x, Y) = f_{y=Y}(x, t) \end{aligned}$$

With  $F_{\text{interior}}(x, y; s)$  and  $F_{y=Y}(x; s)$  denoting the Laplace transforms of  $f_{\text{interior}}(x, y, t)$  and

$f_{y=Y}(x, t)$ , respectively, the Laplace-transformed solution to this system is expressed

$$\Psi = \Psi_1 + \Psi_2$$

$$\Psi_1 = \sum_{\kappa_x} \sin \kappa_x x \cosh \sqrt{\kappa_x^2 + s} y A(s; \kappa_x)$$

with

$$\kappa_x = \frac{\pi}{X}, \frac{2\pi}{X}, \frac{3\pi}{X}, \dots$$

$$A(s; \kappa_x) = \int_0^X dx \sin \kappa_x x \frac{2}{X} M_{\sqrt{\kappa_x^2 + s}} F_{y=Y}(x; s)$$

$$M_{\sqrt{\kappa_x^2 + s}} = \begin{cases} 0 & \text{if } \sqrt{\kappa_x^2 + s} = 0; \\ \frac{1}{\sqrt{\kappa_x^2 + s} \sinh \sqrt{\kappa_x^2 + s} Y} & \text{otherwise.} \end{cases}$$

$$\Psi_2 = \sum_{\kappa_x} \sum_{\kappa_y} \sin \kappa_x x \cos \kappa_y y A(\kappa_x, \kappa_y; s)$$

with

$$\kappa_x = \frac{\pi}{X}, \frac{2\pi}{X}, \frac{3\pi}{X}, \dots$$

$$\kappa_y = 0, \frac{\pi}{Y}, \frac{2\pi}{Y}, \frac{3\pi}{Y}, \dots$$

$$A(\kappa_x, \kappa_y; s) = \int_0^X dx \int_0^Y dy \sin \kappa_x x \frac{2}{X} \cos \kappa_y y N_{\kappa_y} \left( \frac{F_{\text{interior}}(x, y; s) + \Psi(x, y, t=0)}{\kappa_x^2 + \kappa_y^2 + s} \right)$$

$$N_{\kappa_y} = \begin{cases} \frac{1}{Y} & \text{if } \kappa_y = 0; \\ \frac{2}{Y} & \text{otherwise.} \end{cases}$$

This example demonstrates how the logic that solved Example 1 can be retooled to solve Laplace domain problems; the transformed PDE is equivalent to a nonhomogenous Laplace (Poisson) equation with the eigenvalues shifted and the initial condition wedded with the nonhomogeneity.

### Example 6

Wave launched from one end of a rectangular waveguide (homogeneous wave equation in a semi-infinite rectangular waveguide, homogenous Dirichlet conditions on the walls, nonhomogenous Dirichlet on the end face, frequency domain, outgoing wave at infinity):

$$\frac{\partial^2 \Psi}{\partial t^2} = \nabla^2 \Psi$$

$$\Psi(0, y, z) = \Psi(X, y, z) = \Psi(x, 0, z) = \Psi(x, Y, z) = 0,$$

$$\Psi(x, y, 0) = f_{z=0}(x, y, t)$$

With  $F_{z=0}(\omega; x, y)$  denoting the Fourier transforms of  $f_{z=0}(x, y, t)$ , the Fourier transforms of the solution to this system is expressed

$$\Psi = \sum_{\kappa_x} \sum_{\kappa_y} \sin \kappa_x x \sin \kappa_y y e^{-i\sqrt{\omega^2 - \kappa_x^2 - \kappa_y^2} z} A(\omega; \kappa_x, \kappa_y)$$

with

$$\kappa_x = \frac{\pi}{X}, \frac{2\pi}{X}, \frac{3\pi}{X}, \dots$$

$$\kappa_y = \frac{\pi}{Y}, \frac{2\pi}{Y}, \frac{3\pi}{Y}, \dots$$

$$A(\omega; \kappa_x, \kappa_y) = \int_0^X dx \int_0^Y dy \sin \kappa_y x \frac{2}{X} \sin \kappa_y y \frac{2}{Y} F_{z=0}(\omega; x, y)$$

Again, the solution logic of Example 1 is reworked with a shift of the eigenvalues.

## **CHAPTER 6**

### **RECOMMENDATIONS FOR FUTURE DEVELOPMENTS**

To further enhance and build on the success of USFKAD, the following recommendations are offered for future development of this expert system:

1. Offer a choice of output formats (postscript, Mathematics Markup Language)
2. Enable transport of the output to programs like MAPLE, MathCad, or Mathematica for number crunching.
3. Offer graphical supplements to the outputs (such as sample eigenfunction graphs).
4. Develop graphic point-and-click input options to render it more convenient and easy to use.

## REFERENCES

1. Snider, A.D., *Partial Differential Equations: Sources and Solutions*, Prentice-Hall, Upper Saddle River, NJ, 1999.
2. Felsen, L.B., and Marcuvitz, N., *Radiation and Scattering of Waves*, IEEE Press, Piscataway, NJ 1994.
3. Polyanin, A.D., *Linear Partial Differential Equations for Engineers and Scientists*, Chapman and Hall/Chemical Rubber Co., Boca Raton, 2001.
4. Cheb-Terrab, E.S., and von Bulow, K., *A Computational Approach for the Analytical Solving of Partial Differential Equations*, *Computer Physics Communications*, v. 90, 102-116, 1995.
5. Kalnins, E.G. and Miller, Jr., W., *Variable Separable in Mathematical Physics: From Intuitive Concept to Computational Tool*. (Working paper)
6. Wolfram Research, Inc., *Mathematica, Version 5.1*, Champaign, IL (2004).
7. Waterloo Maple, Inc., *Maple V release 3* Canada, 2001.
8. Borland International, *Borland C++ 5.02* Scotts Valley, CA, 1994.

## **APPENDIECES**



**APPENDIX A: COMPARISON OF SOLVING PARTIAL DIFFERENTIAL EQUATIONS USING THE TRADITIONAL METHOD AND USFKAD**

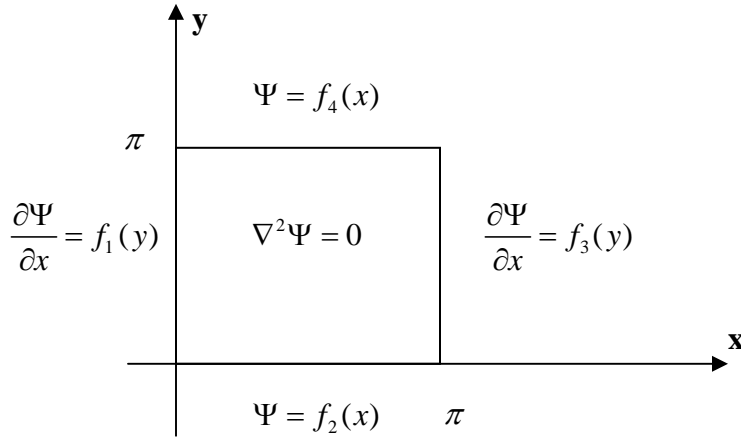
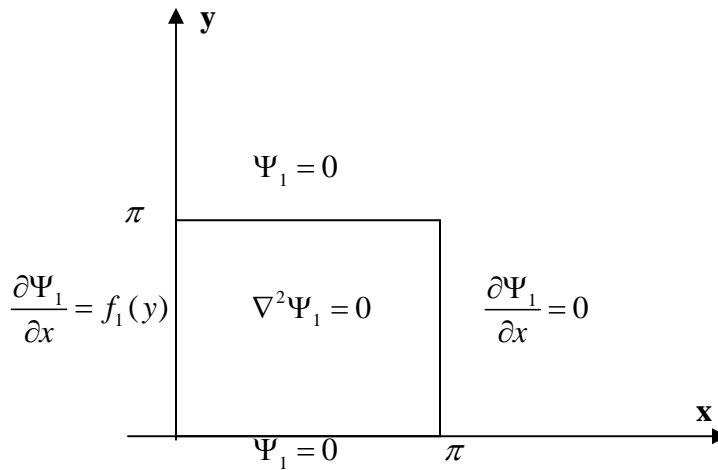


Figure 2: PDE Problem

Consider the boundary value problem shown in Figure 2. We take advantage of the linearity of the equations to simplify the analysis. Suppose we compute the following solutions to the following four sub problems shown in Figure 3 a-d: Decomposition of Problems.



(a)

Figure 3: (a-d) Decomposition of the PDE Problem

Appendix A:  
Continued

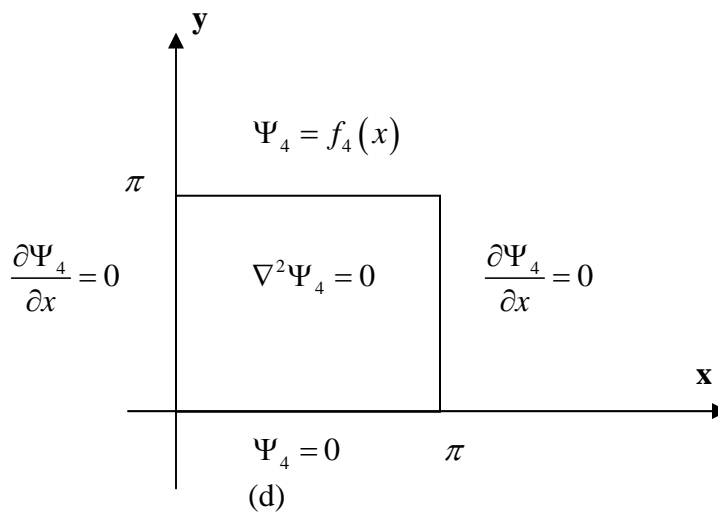
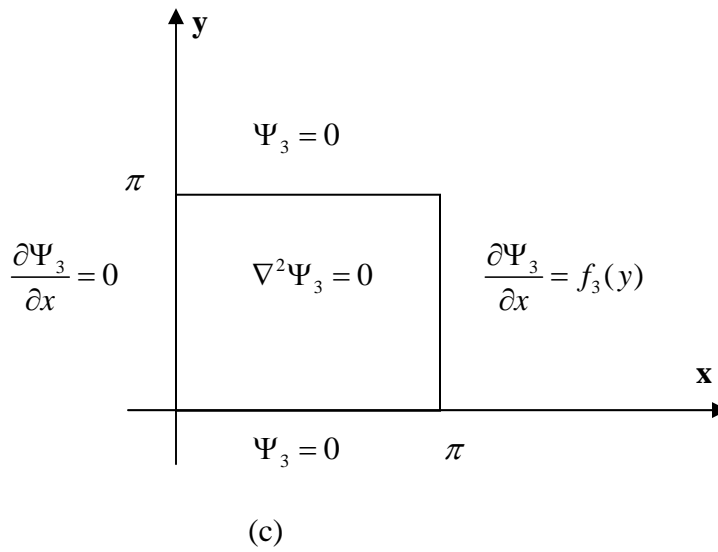
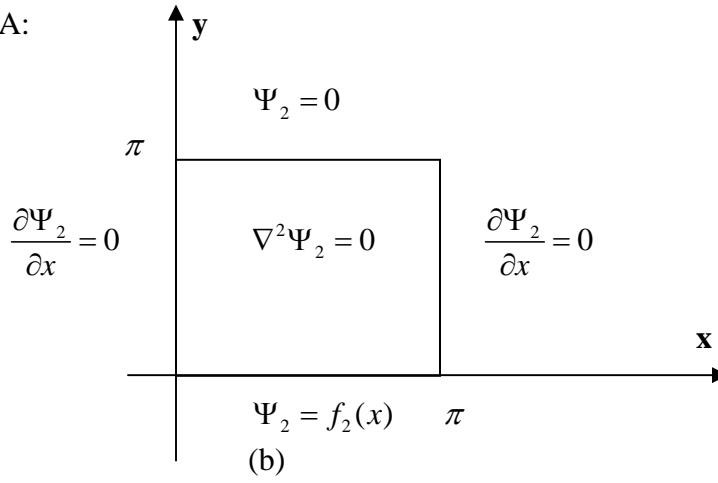


Figure 3: Continued

Appendix A: Continued

Problem 1: Find  $\Psi_1(x, y)$  such that

$$\nabla^2 \Psi_1(x, y) = 0 \text{ inside the square} \quad (1)$$

$$\frac{\partial \Psi_1}{\partial x}(0, y) = f_1(y) \text{ on the left edge } (0 < y < \pi) \quad (2)$$

$$\Psi_1(x, \pi) = 0 \text{ on the top edge } (0 < x < \pi) \quad (3)$$

$$\frac{\partial \Psi_1}{\partial x}(\pi, y) = 0 \text{ on the right edge } (0 < y < \pi) \quad (4)$$

$$\Psi_1(x, 0) = 0 \text{ on the bottom edge } (0 < x < \pi) \quad (5)$$

Problem 2: Find  $\Psi_2(x, y)$  such that

$$\nabla^2 \Psi_2(x, y) = 0 \text{ inside the square} \quad (6)$$

$$\Psi_2(x, 0) = f_2(x) \text{ on the bottom edge } (0 < x < \pi) \quad (7)$$

$$\frac{\partial \Psi_2}{\partial x}(\pi, y) = 0 \text{ on the right edge } (0 < y < \pi) \quad (8)$$

$$\frac{\partial \Psi_2}{\partial x}(0, y) = 0 \text{ on the left edge } (0 < y < \pi) \quad (9)$$

$$\Psi_2(x, \pi) = 0 \text{ on the top edge } (0 < x < \pi) \quad (10)$$

Problem 3: Find  $\Psi_3(x, y)$  such that

$$\nabla^2 \Psi_3(x, y) = 0 \text{ inside the square} \quad (11)$$

$$\frac{\partial \Psi_3}{\partial x}(\pi, y) = f_3(y) \text{ on the right edge } (0 < y < \pi) \quad (12)$$

$$\Psi_3(x, \pi) = 0 \text{ on the top edge } (0 < x < \pi) \quad (13)$$

$$\Psi_3(x, 0) = 0 \text{ on the bottom edge } (0 < x < \pi) \quad (14)$$

## Appendix A: Continued

$$\frac{\partial \Psi_3}{\partial x}(0, y) = 0 \text{ on the left edge } (0 < y < \pi) \quad (15)$$

Problem 4: Find  $\Psi_4(x, y)$  such that

$$\nabla^2 \Psi_4(x, y) = 0 \text{ inside the square} \quad (16)$$

$$\Psi_4(x, \pi) = f_4(x) \text{ on the top edge } (0 < x < \pi) \quad (17)$$

$$\frac{\partial \Psi_4}{\partial x}(\pi, y) = 0 \text{ on the right edge } (0 < y < \pi) \quad (18)$$

$$\Psi_4(x, 0) = 0 \text{ on the bottom edge } (0 < x < \pi) \quad (19)$$

$$\frac{\partial \Psi_4}{\partial x}(0, y) = 0 \text{ on the left edge } (0 < y < \pi) \quad (20)$$

By superposition, we have decomposed the original problem into a set of subproblems in each of which only one boundary condition is nonhomogenous. Let's give as an example how to solve problem 4 by the traditional method.

### **Solution 1**

Solving problem 4:

The solution of Laplace's equation (16) can be expressed as a product of factors.

$$\Psi_4(x, y) = X(x)Y(y) \quad (21)$$

If we substitute (21) into (1) we can get a form where only one variable occurs on each side of the equation:

Appendix A: Continued

$$\begin{aligned}\nabla^2\Psi_4(x, y) = 0 &= \frac{\partial^2}{\partial x^2} X(x)Y(y) + \frac{\partial^2}{\partial y^2} X(x)Y(y) \\ &= X''(x)Y(y) + X(x)Y''(y)\end{aligned}$$

$$\Rightarrow \frac{X''(x)}{X(x)} = -\frac{Y''(y)}{Y(y)} \quad (22)$$

The separated equation (22) implies:

$$\frac{X''(x)}{X(x)} = \lambda \text{ or } X''(x) - \lambda X(x) = 0 \quad (23)$$

$$\text{For some constant } \lambda \text{ (separation constant) and } Y''(y) + \lambda Y(y) = 0 \quad (24)$$

The general solution to the harmonic oscillator equation (23), can be expressed as:

$$X(x) = a_1 \cosh \sqrt{\lambda}x + a_2 \sinh \sqrt{\lambda}x \text{ if } \lambda > 0 \quad (25)$$

$$X(x) = b_1 + b_2x \text{ if } \lambda = 0 \quad (26)$$

$$X(x) = c_1 \cos \sqrt{-\lambda}x + c_2 \sin \sqrt{-\lambda}x \text{ if } \lambda < 0 \quad (27)$$

As for the solutions for (24), they are:

$$Y(y) = d_1 \cos \sqrt{\lambda}y + d_2 \sin \sqrt{\lambda}y \text{ if } \lambda > 0 \quad (28)$$

$$Y(y) = e_1 + e_2y \text{ if } \lambda = 0 \quad (29)$$

$$Y(y) = g_1 \cosh \sqrt{-\lambda}y + g_2 \sinh \sqrt{-\lambda}y \text{ if } \lambda < 0 \quad (30)$$

From (2)-(5):

$$X'(\pi) = 0, X'(0) = 0 \quad (31)$$

$$Y(0) = 0 \quad (32)$$

Appendix A: Continued

$$X'(x) = \sqrt{\lambda}a_1 \sinh \sqrt{\lambda}x + \sqrt{\lambda}a_2 \cosh \sqrt{\lambda}x \text{ if } \lambda > 0 \quad (33)$$

$$X'(x) = b_2 \text{ if } \lambda = 0 \quad (34)$$

$$X'(x) = -\sqrt{-\lambda}c_1 \sin \sqrt{-\lambda}x + \sqrt{-\lambda}c_2 \cos \sqrt{-\lambda}x \text{ if } \lambda < 0 \quad (35)$$

Applying the boundary condition  $X'(0) = 0$ , we get

$$X'(0) = \sqrt{\lambda}a_2 = 0 \text{ if } \lambda > 0 \quad (36)$$

$$X'(0) = b_2 = 0 \text{ if } \lambda = 0 \quad (37)$$

$$X'(0) = \sqrt{-\lambda}c_2 = 0 \text{ if } \lambda < 0 \quad (38)$$

$$\text{So } X(x) = a_1 \cosh \sqrt{\lambda}x \text{ if } \lambda > 0 \quad (39)$$

$$X(x) = b_1 \text{ if } \lambda = 0 \quad (40)$$

$$X(x) = c_1 \cos \sqrt{-\lambda}x \text{ if } \lambda < 0 \quad (41)$$

Imposing  $X'(\pi) = 0$  implies:

$$X'(\pi) = \sqrt{\lambda}a_1 \sinh \sqrt{\lambda}\pi = 0 \text{ if } \lambda > 0 \quad (42)$$

$$X'(\pi) = 0 \text{ if } \lambda = 0 \quad (43)$$

$$X'(\pi) = \sqrt{-\lambda}c_1 \sinh \sqrt{-\lambda}\pi = 0 \text{ if } \lambda < 0 \quad (44)$$

Since we are not interested in trivial solutions,  $a_1, b_1$ , or  $c_1$  can not equal zero, so we need to satisfy (42)-(44) by the selection of  $\lambda$ . The choice  $\lambda = 0$  is acceptable  $\Rightarrow X(x) = b_1$  (=constant). No positive value for  $\lambda$  yield solutions, because the sin function in (42) never vanishes. However, the sin function vanishes whenever  $\sqrt{-\lambda} = n = 1, 2, 3, \dots$ , and the corresponding solution (39)-(41) is  $c_1 \cos nx$ . So we have

Appendix A: Continued

nontrivial solutions for the x factor in (21) if  $\lambda = 0, -1, -4, \dots, -n^2, \dots$ , applying (32) to (28)-(30):

$$Y(0) = d_1 = 0 \text{ if } \lambda > 0$$

$$Y(0) = e_1 = 0 \text{ if } \lambda = 0$$

$$Y(0) = g_1 = 0 \text{ if } \lambda < 0$$

$$\Rightarrow Y_n(y) = g_2 \sinh ny \text{ and } Y_0(y) = e_2 y$$

$$\Rightarrow \phi_n = \cos nx \sinh ny$$

$$\Rightarrow \Psi_4(x, y) = a_0 y + \sum_{n=1}^{\infty} a_n \cos nx \sinh ny$$

$$\text{Where } a_0 = \frac{1}{\pi^2} \int_0^{\pi} f_4(x) dx, a_n > 0 = \frac{2}{\pi \sinh n\pi} \int_0^{\pi} f_4(x) \cos nx$$

You see how long and cumbersome it was to do only problem 4. As for problems 3, 2, and 1, we will get the following after a lot of manipulation:

Problem 3:

$$\Psi_3(x, y) = \sum_{n=1}^{\infty} b_n \cosh nx \sin ny \text{ where } b_n = \frac{2}{n\pi \sinh n\pi} \int_0^{\pi} f_3(y) \sin ny dy$$

Problem 2:

$$\Psi_2(x, y) = c_0(\pi - y) + \sum_{n=1}^{\infty} c_n \cos nx \sinh n(\pi - y) \text{ where}$$

Appendix A: Continued

$$c_0 = \frac{1}{\pi^2} \int_0^{\pi} f_2(x) dx$$

$$c_{n>0} = \frac{2}{\pi \sinh n\pi} \int_0^{\pi} f_2(x) \cos nxdx$$

Problem 1:

$$\Psi_1(x, y) = \sum_{n=1}^{\infty} d_n \cosh n(\pi - x) \sin ny \text{ where}$$

$$d_n = \frac{-2}{n\pi \sinh n\pi} \int_0^{\pi} f_1(y) \sin ny dy..$$

The final answer will be

$$\Psi(x, y) = \Psi_1(x, y) + \Psi_2(x, y) + \Psi_3(x, y) + \Psi_4(x, y).$$

**Solution 2**

Complete solution using USFKAD.

$$\Psi = \Psi_1 + \Psi_2 + \Psi_3 + \Psi_4$$

$$\Psi_1 = \sum_{\kappa_y} \sin \kappa_y y \cosh \kappa_y (X - x) A(\kappa_y)$$

where

$$\kappa_y = \frac{\pi}{Y}, \frac{2\pi}{Y}, \frac{3\pi}{Y}, \dots$$

$$A(\kappa_y) = \int_0^Y dy \sin \kappa_y y \frac{2}{Y} M_{\kappa_y} f_{x=0}(y)$$



Appendix A: Continued

$$M_{\kappa_y} = \begin{cases} 0 & \text{if } \kappa_y = 0; \\ \frac{1}{\kappa_y \sinh \kappa_y X} & \text{otherwise.} \end{cases}$$

$$\Psi_2 = \sum_{\kappa_x} \cos \kappa_x x \eta_y(y; \kappa_x) A(\kappa_x)$$

where

$$\kappa_x = 0, \frac{\pi}{X}, \frac{2\pi}{X}, \frac{3\pi}{X}, \dots$$

$$\eta_y(y; \kappa_x) = \begin{cases} Y - y & \text{if } \kappa_x = 0 \\ \sinh \kappa_x (Y - y) & \text{otherwise.} \end{cases}$$

$$A(\kappa_x) = \int_0^X dx \cos \kappa_x x N_{\kappa_x} M_{\kappa_x} f_{y=0}(x)$$

$$N_{\kappa_x} = \begin{cases} \frac{1}{X} & \text{if } \kappa_x = 0; \\ \frac{2}{X} & \text{otherwise.} \end{cases}$$

$$M_{\kappa_x} = \begin{cases} \frac{1}{Y} & \text{if } \kappa_x = 0; \\ \frac{1}{\sinh \kappa_x Y} & \text{otherwise.} \end{cases}$$

$$\Psi_3 = \sum_{\kappa_y} \sin \kappa_y y \cosh \kappa_y x A(\kappa_y)$$

where

$$\kappa_y = \frac{\pi}{Y}, \frac{2\pi}{Y}, \frac{3\pi}{Y}, \dots$$

$$A(\kappa_y) = \int_0^Y dy \sin \kappa_y y \frac{2}{Y} M_{\kappa_y} f_{x=X}(y)$$

$$M_{\kappa_y} = \begin{cases} 0 & \text{if } \kappa_y = 0; \\ \frac{1}{\kappa_y \sinh \kappa_y X} & \text{otherwise.} \end{cases}$$

$$\Psi_4 = \sum_{\kappa_x} \cos \kappa_x x \eta_y(y; \kappa_x) A(\kappa_x)$$

where

Appendix A: Continued

$$\kappa_x = 0, \frac{\pi}{X}, \frac{2\pi}{X}, \frac{3\pi}{X}, \dots$$

$$\eta_y(y; \kappa_x) = \begin{cases} y & \text{if } \kappa_x = 0 \\ \sinh \kappa_x y & \text{otherwise.} \end{cases}$$

$$A(\kappa_x) = \int_0^X dx \cos \kappa_x x N_{\kappa_x} M_{\kappa_x} f_{y=Y}(x)$$

$$N_{\kappa_x} = \begin{cases} \frac{1}{X} & \text{if } \kappa_x = 0; \\ \frac{2}{X} & \text{otherwise.} \end{cases}$$

$$M_{\kappa_x} = \begin{cases} \frac{1}{Y} & \text{if } \kappa_x = 0; \\ \frac{1}{\sinh \kappa_x Y} & \text{otherwise.} \end{cases}$$

If you set  $X=Y=\pi$ , you will find that both the traditional method and USFKAD solutions are identical.

## APPENDIX B: READ ME

Welcome to USFKAD, the software for solving partial differential equations analytically, by separation of variables. If you find this software is of value to you, please consider making a donation to USF students via the Allen Gondeck scholarship fund, through Prof. A. D. Snider, University of South Florida, ENB 118, Tampa FL 33620. (Thank you.)

The task of constructing complete solutions by separation of variables is quite tedious, and the software can do this for you only if you follow the format/notation conventions precisely.

The current version handles (homogeneous or nonhomogeneous) (mixed) Dirichlet, Neumann, constant-coefficient Robin boundary conditions, or singular boundary conditions for the (possibly) nonhomogeneous Poisson, diffusion, or wave equations, in the time, frequency, or Laplace domains.

You will have to label the dimensions of your domain to conform to one of the following conventions:

1.  $0 < x, y, z, \theta < X, Y, Z, \Theta$
2.  $0 < x, y, z < \infty$  (Do not use  $-\infty < x, y, z < X, Y, Z$ )
3.  $-\infty < x, y, z < \infty$
4.  $0 < \theta < 2\pi$  (periodic)
5.  $0 < a < r < b < \infty$

Appendix B: Continued

6.  $0 < a < r < \infty$
7.  $0 < r < b < \infty$
8.  $0 < r < \infty$
9.  $0 < \theta < 2\pi$  and  $0 < \phi < \pi$  (spherical coordinates)

Dirichlet boundary conditions take the form

$$\psi(0,y,z) = f_{x=0}(y,z) ; \psi(X,y,z) = f_{x=X}(y,z)$$

(and similarly for  $y$ ,  $z$ ,  $r$ , and  $\theta$ ).

Neumann boundary conditions take the form

$$\partial\psi(0,y,z)/\partial x = f_{x=0}(y,z) ; \partial\psi(X,y,z)/\partial x = f_{x=X}(y,z)$$

(and similarly for  $y$ ,  $z$ ,  $r$ ,  $\rho$ , and  $\theta$ ). **Note that the relevant partial will *not*, in general, be the *external* normal derivative.**

Robin boundary conditions take the form

$$\alpha_{x=0} \psi(0,y,z) + \partial\psi(0,y,z)/\partial x = f_{x=0}(y,z) ;$$

$$\alpha_X \psi(X,y,z) + \partial\psi(X,y,z)/\partial x = f_{x=X}(y,z)$$

(and similarly for  $y$ ,  $z$ ,  $r$ ,  $\rho$ , and  $\theta$ ). **Note that the relevant partial will *not*, in general, be the *external* normal derivative.** The coefficient  $\alpha$  is presumed constant.

## Appendix B: Continued

To use the software, doubleclick on the .exe file, and follow the menu instructions *carefully*. Give the name of your output file a .tex subscript. The software does not alert the user if inconsistent parameters are input - it simply fails to produce an output file.

Run *latex* on the output file, and either view the .dvi result or *dvips* it to postscript, and print out. You may make format changes to the .tex file if you wish.

Please inform A. D. Snider by email ([snider@eng.usf.edu](mailto:snider@eng.usf.edu)) if you feel the software has returned an incorrect answer, or if you desire elaboration of the answer; include a complete problem statement and the output .tex file, and your comments.

Enjoy!

Sami Kadamani

Dave Snider

## APPENDIX C: ToUseUSFKAD.pdf

### To use USFKAD:

Create a folder on your hard drive, place it in root of the drive, and name it USFKAD. If you have several hard drives, you may use any drive you want. For this tutorial, d-drive is used.

Location of the folder, will be `d:\USFKAD>`

Open Windows Explorer, browse to `d:\USFKAD` and double click USFKAD program file.

The Command Prompt will open, and you get a warning message.

Click OK to proceed.

Now you have to create the output file, and in this example it is named: `testing.tex`  
Remember the extension `tex`

Type in your filename, in this case `testing.tex`

Follow the instructions in USFKAD, and for this tutorial we used the following input:

`0, 0, 2, 0, 1, 1, 2 and 2`

This will create the file `testing.tex` in the folder `d:\USFKAD`

Next step will be to create the dvi file. Click on Start, Run..

Type `cmd` in the Run window. This will open the command prompt.

In the command prompt, browse to your folder. "`d:`" will take you to the d drive, then `cd usfkad`, will change the directory to where the `testing.tex` file is.

Type "`d: cd usfkad`" to get to d drive, and the usfkad folder

Once you are in the correct folder, type `latex` to open the application. Then press enter to proceed after the welcome note.

Open LaTeX by typing `latex`

Type the file name, in this case `testing`. There is no need for the file extension, but it has to be `tex` when you created it.

Type the filename, in this case `testing`

This will give you a dvi output file. If there are any problems, you will see it on this screen

The status of creating the dvi file

Appendix C: Continued

Next step will be to view dvi file. Click on Start, Run..

Type “yap” in the Run window. The program used to view the dvi file.  
In Yap window, choose open file, and select testing.dvi

Open the testing.dvi file.  
Yap will show the final output

Output from the testing.tex file crated with USFKAD

### **ABOUT THE AUTHOR**

Sami M. Kadamani is currently an instructor at Hillsborough Community College in Tampa, Florida. He completed his Bachelor of Science in Electrical Engineering from the University of Delaware in 1983, and his Master of Science in Electrical Engineering from North Carolina A & T State University in 1986. In 1986 Mr. Kadamani moved to Tampa, Florida to pursue his Doctoral of Philosophy degree in Engineering Science at the University of South Florida, which he completed in May 2005.