Harmonic Function Theory and Mathematica

Sheldon Axler
Mathematics Department
Michigan State University
East Lansing, MI 48824 USA

Paul Bourdon
Mathematics Department
Washington and Lee University
Lexington, VA 24450 USA

Wade Ramey
Mathematics Department
Michigan State University
East Lansing, MI 48824 USA

15 December 1996

Contents

Getting Started

These notes constitute the manual for our computer software dealing with harmonic functions. Using *Mathematica*,* a symbolic processing program, we have written routines to compute many of the expressions that arise in the study of harmonic functions. These routines allow the user to make symbolic calculations that would take a prohibitive amount of time if done without a computer. For example, the Poisson integral of any polynomial can be computed exactly. Our book *Harmonic Function Theory* [ABR] was the starting point for most of the algorithms used in this software; we have also made use of the other publications listed in the References section of these notes.

A copy of *Mathematica* 2.0 (or a later version) is needed to make use of our software package, which will work on any computer that runs *Mathematica*.

Our software package, called *HFT* (for *Harmonic Function Theory*), and this documentation are electronically available free of charge. To obtain the most recent versions, send an e-mail request to `axler@math.msu.edu` or retrieve them using a World Wide Web browser from `http://math.msu.edu/~axler`.

Comments, suggestions, and bug reports about the software, this document, and the book are welcome; please send them to `axler@math.msu.edu`.

Once you have electronically received our software package, cut out any extraneous lines at the beginning or end and save the resulting file on your computer in a text only (ASCII) file with a name appropriate for your computer (for example, HFT.m).

A typical interactive *Mathematica* session consists of input typed by the user and output displayed by the computer. In the examples below, the input typed by the user is preceded by *In[m] :=*, where m is the line number in that session. Output displayed by the computer is preceded by *Out[m] =*.

To begin a session with our package, first start the *Mathematica* program. Then, assuming that you used the name HFT.m for our package, type the following command:

In[1] := `<<HFT.m`

(You may need to specify the directory or folder in which HFT.m is located. With some versions of *Mathematica*, you can type `<<` and then use the Paste File Pathname command, located in the Prepare Input submenu under the Action menu, to paste the full pathname of HFT.m into your input cell.) You are now ready to use the functions described below.

On-line Help

These notes provide the only complete documentation for using our software package. However, users can be reminded about the basic use and syntax of each function created by our software package by entering `?` followed by the function name:

In[1] := `?Dirichlet`

```
Dirichlet[f, x] gives the harmonic function that equals f on the unit sphere.
        Here f must be a polynomial function of x.
```

The *Mathematica* statement `Names[ "HFT`*"]` will produce a complete list of the functions created by our software package.

---

* *Mathematica* is a registered trademark of Wolfram Research.

With some versions of *Mathematica*, you can use the Function Browser to obtain help with the functions created by our software package. To do so, open the Function Browser, select Loaded Packages, and then select the name of this package.

Calculus in $\mathbf{R}^n$

Norm and ExpandNorm

`Norm[x]` is the Euclidean norm of a vector x:

In[1] := `Norm[5x]`

Out[1] = $5\,|x|$

Although `Norm[x]` must be used as the input form, the example above shows that our package uses the more standard notation $|x|$ as the output form.

`ExpandNorm` expands norms of sums (and differences) by replacing expressions of the form $|x{+}y|$ with $(|x|^2{+}|y|^2{+}2x{\cdot}y)^{1/2}$. `ExpandNorm` is often useful for verifying identities. For example, to prove the Symmetry Lemma (1.10 in [ABR]), which states that $\left|\frac{y}{|y|} - |y|x\right| = \left|\frac{x}{|x|} - |x|y\right|$ for all nonzero $x, y \in \mathbf{R}^n$, we subtract the left-hand side of the alleged equality from the right-hand side, hoping that the result equals 0:

In[2] := `Norm[ x/Norm[x] - Norm[x]y ] - Norm[ y/Norm[y] - Norm[y]x ]`

Out[2] = $|\frac{1}{|x|}\,x - |x|\,y| - |\frac{1}{|y|}\,y - |y|\,x|$

The last output does not look like 0, so we apply `ExpandNorm` to it. The symbol `%` used below is the *Mathematica* abbreviation for the last output:

In[3] := `ExpandNorm[%]`

Out[3] = 0

Thus the Symmetry Lemma has been proved.

Partial

`Partial[ f, x[j] ]` is the partial derivative of f with respect to x_j. Here x denotes a vector in Euclidean space, and x_j is the j^{th} coordinate of x. The advantage of using `Partial` rather than the built-in *Mathematica* function `D` is that `Partial` can deal with norms and dot products. Note that the dot product of b and x is denoted by `b.x`:

In[1] := `Partial[ Norm[a + x]^(b.x) , x[5] ]`

Out[1] = $b_5\,\mathrm{Log}[|a+x|]\,|a+x|^{b.x} + b.x\,|a+x|^{-2+b.x}\,(a_5 + x_5)$

Although you must use `x[5]` as the input form to denote the fifth coordinate of x, the example above shows that the more standard mathematical notation x_5 is used as the output form. If you prefer to have coordinates in output appear in the form $x[j]$ instead of x_j, see the *Subscripts* subsection later in these notes.

For taking multiple partial derivatives, `Partial` uses the same syntax as the built-in *Mathematica* function `D`. For example, the following command computes the partial derivative of $|x|$ twice with respect to the first coordinate and three times with respect to the fourth coordinate:

In[2] := `Partial[ Norm[x], {x[1], 2}, {x[4], 3} ]`

$$Out[2] = \frac{3\,(3\,|x|^4\,x_4 - 15\,|x|^2\,{x_1}^2\,x_4 - 5\,|x|^2\,{x_4}^3 + 35\,{x_1}^2\,{x_4}^3)}{|x|^9}$$

Laplacian

`Laplacian[f, x]` is the Laplacian of f with respect to x:

In[1] := `Laplacian[ x[3] Norm[x], x ]`

$$Out[1] = \frac{x_3 + \mathtt{Dimension}[x]\,x_3}{|x|}$$

Note that in the last result, the Laplacian depends upon `Dimension[x]`, which is the dimension of the Euclidean space in which x lives. For work in $\mathbf{R}^n$, `SetDimension[x, n]` should be used to set `Dimension[x]` to equal n; here n can be a symbol or a specific integer, say 8. We illustrate this procedure by showing that for each z with $|z| = 1$, the Poisson kernel $P(\cdot, z)$ is harmonic (in other words, what follows is a proof of Proposition 1.15 of [ABR]):

In[2] := `SetDimension[x, n]`

In[3] := `Norm[z] = 1 ;`

In[4] := `Laplacian[ (1 - Norm[x]^2)/Norm[x - z]^n, x ]`

$$Out[4] = 2\,|x - z|^{-2-n}\,(n - 2\,n\,x.z + n\,|x|^2 - n\,|x - z|^2)$$

In[5] := `ExpandNorm[%]`

Out[5] = `0`

Thus the Laplacian of $P(\cdot, z)$ is 0, and hence $P(\cdot, z)$ is harmonic.

The semi-colon at the end of *In[3]* suppresses that statement's output, which there is no need to see.

`(Laplacian^m)` can be used, with the same syntax as `Laplacian`, to evaluate the m^{th} power of the Laplacian. For example, here we find the biLaplacian (the Laplacian of the Laplacian) of $1/|x|$ on $\mathbf{R}^n$:

In[6] := `(Laplacian^2)[ 1/Norm[x], x ]`

Out[6] = $\frac{3\,(15 - 8\,\mathtt{n} + \mathtt{n}^2)}{|\mathtt{x}|^5}$

If x and y are symbols, then `Laplacian[f, x, y]` gives the Laplacian of f with respect to the vector (x, y), where x is thought of as a vector, y is thought of as a real variable, and (x, y) is thought of as a vector in a Euclidean space whose dimension is one more than the dimension of x. This format is often useful when working with functions defined on the upper half-space, as in Chapter 7 of [ABR]. Note that if we are thinking of (x, y) as an element of $\mathbf{R}^n$, then x should be defined to have dimension $n-1$. We illustrate the use of this format of the `Laplacian` command by showing that for each $t \in \mathbf{R}^{n-1}$, the Poisson kernel $P_H(\cdot, t)$ for the upper half-space is harmonic:

In[7] := `SetDimension[ x, n-1 ]`

In[8] := `Laplacian[ y / (Norm[x - t]^2 + y^2)^(n/2), x, y ]`

Out[8] = `0`

Thus the Laplacian of $P_H(\cdot, t)$ is 0, and hence $P_H(\cdot, t)$ is harmonic.

Gradient

`Gradient[f, x]` is the gradient of f with respect to the vector x:

In[1] := `Gradient[ a.x + x[k] + Norm[x], x ]`

Out[1] = $\mathtt{a} + \mathtt{Delta}[\mathtt{k}] + \frac{1}{|\mathtt{x}|}\,\mathtt{x}$

In the last output, `Delta[k]` denotes the vector that equals 1 in the k^{th} coordinate and 0 in the other coordinates.

NormalD

`NormalD[f, z]` equals $(D_{\mathbf{n}} f)(z)$, the outward normal derivative of f at a point z on the unit sphere (the boundary of the unit ball).

In the section of Chapter 1 of [ABR] titled *The Poisson Kernel for the Ball*, we derived the formula for the Poisson kernel by showing that $P(x, z) = (2-n)^{-1}(D_{\mathbf{n}} v)(z)$, where x is a point in the unit ball and v is the function defined by

$$v(z) = |z - x|^{2-n} - |x|^{2-n}|z - \frac{x}{|x|^2}|^{2-n}.$$

We left the computation of $D_{\mathbf{n}} v$, which is a bit complicated, to the reader. Now we show how this computation can be done easily by a computer. We begin by defining v as above:

In[1] := `SetDimension[z, n]`

In[2] :=
```
v[z] =
 Norm[z - x]^(2 - n) -
 Norm[x]^(2-n) Norm[ z - x/Norm[x]^2 ]^(2-n) ;
```

Now we take the normal derivative of v:

In[3] := `NormalD[ v[z], z ]`

$$Out[3] = \frac{(2-\mathtt{n})\,(1-\mathtt{x.z})}{|-\mathtt{x}+\mathtt{z}|^{\mathtt{n}}} + \frac{(2-\mathtt{n})\,(\mathtt{x.z}-|\mathtt{x}|^2)}{|\mathtt{x}|^{\mathtt{n}}\,\left|\mathtt{z}-|\mathtt{x}|^{-2}\,\mathtt{x}\right|^{\mathtt{n}}}$$

In the following command, we ask the computer to replace $|z - |x|^{-2}\,x|$ in the last output with $|z-x|/|x|$; the Symmetry Lemma (1.10 of [ABR]) implies that the two expressions are equal. The *Mathematica* syntax `/.  f -> g` means that `f` should be replaced by `g`:

In[4] := `% /.  Norm[ z - Norm[x]^(-2) x ] -> Norm[z - x]/Norm[x]`

$$Out[4] = \frac{2-\mathtt{n}-2\,|\mathtt{x}|^2+\mathtt{n}\,|\mathtt{x}|^2}{|-\mathtt{x}+\mathtt{z}|^{\mathtt{n}}}$$

Recall that the Poisson kernel equals $(2-n)^{-1}(D_{\mathbf{n}}v)(z)$. Dividing the last output, which equals $(D_{\mathbf{n}}v)(z)$, by $(2-n)$ thus gives the desired expression for the Poisson kernel:

In[5] := `% / (2-n)`

$$Out[5] = \frac{1-|\mathtt{x}|^2}{|-\mathtt{x}+\mathtt{z}|^{\mathtt{n}}}$$

Divergence

`Divergence[f, x]` is the divergence of f with respect to the vector x:

In[1] := `SetDimension[x, n]`

In[2] := `Divergence[ x/Norm[x]^2, x ]`

$$Out[2] = \frac{-2+\mathtt{n}}{|\mathtt{x}|^2}$$

Matrices

Mathematica uses `.` to represent matrix multiplication as well as dot product. When using our software package, you must use the `SetDimension` command to tell *Mathematica* whenever a symbol is being thought of as a matrix. To think of A as an m-by-n matrix, enter the command `SetDimension[ A, {m, n} ]`. For the examples in this section, we want to think of A as a square matrix, so we make the two dimensions of A equal:

In[1] := `SetDimension[ A, {n, n} ]`

In[2] := `Gradient[ Norm[A.x], x ]`

Out[2] = $\frac{1}{|\mathtt{A.x}|}\,(\mathtt{A.x}).\mathtt{A}$

To interpret input or output involving products of matrices and vectors, think of vectors as either column vectors or row vectors, whichever makes sense in context. Thus in the input above, x is obviously a vector, because we are taking the gradient of some function with respect to x. For $A.x$ to make sense as a vector in $\mathbf{R}^n$, we must think of x as a column vector (an n-by-1 matrix). To interpret $(A.x).A$ in the last output, we again think of x as a column vector (an n-by-1 matrix), so that $A.x$ makes sense as an element of $\mathbf{R}^n$. Because $A.x$ is then multiplied by the n-by-n matrix A, we must think of $A.x$ as a row vector (an 1-by-n matrix), so that $(A.x).A$ is a vector, as expected here (because it is the gradient of a function).

In the example above we did not enter the command `SetDimension[x, n]` to tell *Mathematica* that x is a vector (although it would not have hurt to do so). Our software package can almost always use the context to distinguish scalars from vectors (if both interpretations make sense, the package assumes that a symbol represents a scalar). However, you must always explicitly use the `SetDimension` command to tell *Mathematica* which objects are matrices.

If A is a matrix, then $A_{j,.}$ denotes the j^{th} row of A and $A_{.,j}$ denotes the j^{th} column of A:

In[3] := `Partial[ Norm[A.x] + 5 Norm[x.A], x[3] ]`

Out[3] = $\frac{(\mathtt{A.x}).\mathtt{A}_{.,3}}{|\mathtt{A.x}|} + \frac{5\,(\mathtt{x.A}).\mathtt{A}_{3,.}}{|\mathtt{x.A}|}$

Note that in the input above, x is thought of as a column vector in the term $A.x$ and as a row vector in the term $x.A$. To interpret $(A.x).A_{.,3}$ in the last output, we think of x as a column vector, so that $A.x$ is a vector. Then $(A.x).A_{.,3}$ is just the dot product of two vectors, so that it is a number, as expected here.

The next example illustrates two more functions involving matrices. `Transpose[A]` is the matrix obtained from A by interchanging the rows and columns. The Hilbert-Schmidt norm of A, denoted $|A|_2$, is the square root of the sum of the squares of all entries of A.

In[4] := `Laplacian[ Norm[x.A], x ]`

Out[4] = $\frac{|\mathtt{A}|_2{}^2\,|\mathtt{x.A}|^2 - |(\mathtt{x.A}).\mathtt{Transpose[A]}|^2}{|\mathtt{x.A}|^3}$

If you turn the `Subscripts` feature off (see the *Subscripts* subsection later in these notes), then instead of $\mathtt{A}_{j,.}$, $\mathtt{A}_{.,j}$, and $|\mathtt{A}|_2$ you will see `A[j, .]`, `A[., j]`, and `HilbertSchmidt[A]`. For input use `A[j, "."]`, `A[".", j]`, and `HilbertSchmidt[A]`.

`Tr[A]` is the trace of a matrix A:

In[5] := `Divergence[ A.x/Norm[x], x ]`

Out[5] = $-(\frac{\mathtt{x.(A.x)}}{|\mathtt{x}|^3}) + \frac{\mathtt{Tr[A]}}{|\mathtt{x}|}$

Jacobian

`Jacobian[f, x]` is the Jacobian derivative of f with respect to x. Here f should be a function of x taking values in some Euclidean space, and the Jacobian derivative is the usual matrix consisting of partial derivatives of the coordinates of f:

In[1] := `SetDimension[x, n]`

In[2] := `Jacobian[ x/Norm[x]^2, x ]`

Out[2] = $\frac{-2}{|\mathtt{x}|^4}\mathtt{Transpose[\{x\}].\{x\}} + |\mathtt{x}|^{-2}\,\mathtt{IdentityMatrix[n]}$

Although x can be thought of as a row vector or a column vector, depending on the context, if we were using explicit coordinates we would represent x in *Mathematica* by something like `{x1, x2, ... , xn}`. Thus `{x}` would be `{{x1, x2, ... , xn}}`, which is a 1-by-n matrix. Thus `Transpose[{x}]` is an n-by-1 matrix, and the product `Transpose[{x}].{x}`, which appears in the last output, is an n-by-n matrix, as expected.

`IdentityMatrix[n]`, which also appears in the last output, denotes the n-by-n matrix with 1's on the diagonal and 0's elsewhere.

Homogeneous

`Homogeneous[u, m, x]` is the term of degree m in the homogeneous expansion of u at the origin; here u is thought of as a function of the vector x:

In[1] := `SetDimension[x, 3]`

In[2] := `Homogeneous[ (1 - Norm[x]^2)/Cos[ x[1] ]^(3/2), 4, x ]`

Out[2] = $\frac{-11\,x_1{}^4 - 24\,x_1{}^2\,x_2{}^2 - 24\,x_1{}^2\,x_3{}^2}{32}$

An alternate form of `Homogeneous` allows the user to find homogeneous expansions about points other than the origin. `Homogeneous[u, m, x, b]` is the term of degree m in the homogeneous expansion of u, as a function of x, about b:

In[3] := `Homogeneous[ x[1] x[2] + x[3]^4, 2, x, b ]`

Out[3] = $(-b_1 + x_1)\,(-b_2 + x_2) + 6\,b_3{}^2\,(-b_3 + x_3)^2$

The first time in each *Mathematica* session that you use `Homogeneous` to find an expansion about a point other than the origin, you may see a message that `Togetherness` has been turned off. The *Togetherness* subsection later in these notes provides an explanation for this message.

Taylor

`Taylor[u, m, x]` is the sum of all terms of degree at most m in the Taylor series expansion of u at the origin; here u is thought of as a function of the vector x:

In[1] := `Taylor[ x[1]^2 / (6 + x[2] + Cos[ x[3] ]), 4, x ]`

Out[1] = $\dfrac{98\,x_1{}^2 - 14\,x_1{}^2\,x_2 + 2\,x_1{}^2\,x_2{}^2 + 7\,x_1{}^2\,x_3{}^2}{686}$

An alternate form of `Taylor` allows the user to find Taylor expansions about points other than the origin. `Taylor[u, m, x, b]` is the sum of all terms of degree at most m in the Taylor series expansion of u, as a function of x, about b:

In[2] := `Taylor[ 1 + x[1] x[2] + x[1]^2, 2, x, b ]`

Out[2] = $1 + b_1{}^2 + b_1\,b_2 + (2\,b_1 + b_2)\,(-b_1 + x_1) + (-b_1 + x_1)^2 +$
$b_1\,(-b_2 + x_2) + (-b_1 + x_1)\,(-b_2 + x_2)$

The first time in each *Mathematica* session that you use `Taylor` to find an expansion about a point other than the origin, you may see a message that `Togetherness` has been turned off. The *Togetherness* subsection later in these notes provides an explanation for this message.

Non-explicit Functions

Our examples so far have involved only concretely defined functions. But functions in symbolic form can also be used with each of our differentiation commands. Here, for example, is how to find the gradient of the function whose value at x is $f\Big(\big(g(3x)\big)^2\Big)$:

In[1] := `Gradient[ f[ g[3x]^2 ], x ]`

Out[1] = 6 g[3 x] f′[g[3 x]^2] Gradient[g][3 x]

Here `Gradient[g][3 x]` denotes the gradient of g, evaluated at $3x$. For the example above to make sense, we (and the computer) must think of g as a real-valued function on some Euclidean space (where x lives) and f as a function from **R** to **R**. Let's find the Laplacian of the same function:

In[2] := `Laplacian[ f[ g[3x]^2 ], x ]`

Out[2] = 18(|Gradient[g][3 x]|^2 + g[3 x] Laplacian[g][3 x]) f′[g[3 x]^2] +
36 g[3 x]^2 |Gradient[g][3 x]|^2 f″[g[3 x]^2]

`Laplacian[g][3x]` denotes, of course, the Laplacian of g, evaluated at $3x$.

All our differentiation commands (`Partial`, `Laplacian`, `Gradient`, `NormalD`, `Divergence`, `Jacobian`) and `Homogeneous` and `Taylor` can be used with non-explicit functions. For example, if working in $\mathbf{R}^2$, to find the term of degree 3 in the homogeneous expansion of $f\Big(\big(g(3x)\big)^2\Big)$ about the origin, you would enter the command `SetDimension[x, 2]` and then the command `Homogeneous[ f[g[3x]^2], 3, x ]`. Try it! In the answer you will see terms such as $\texttt{Partial}_{1,1,2}[\mathrm{g}][0]$, which denotes the partial derivative of g, twice with respect to the first variable and once with respect to the second variable, evaluated at 0.

Vector-valued Functions

Suppose we compute the partial derivative with respect to x_1 of the function that takes x to $f\Big(g\big(h(x)^2\big)\Big)$:

In[1] := `Partial[ f[ g[ h[x]^2 ] ], x[1] ]`

Out[1] = $2\,\mathrm{h}[\mathrm{x}]\,\mathrm{f}'[\mathrm{g}[\mathrm{h}[\mathrm{x}]^2]]\,\mathrm{g}'[\mathrm{h}[\mathrm{x}]^2]\,\texttt{Partial}_1[\mathrm{h}][\mathrm{x}]$

The example above illustrates a general principle: Our package assumes that all functions are real valued, unless told otherwise. Thus the last output is correct if we are thinking of g as a real-valued function. If we want to think of g as taking values in $\mathbf{R}^n$, we must first enter the command `SetDimension[ g[ _ ], n ]`, which instructs the computer that g is a function with range in $\mathbf{R}^n$:

In[2] := `SetDimension[ g[ _ ], n ]`

In[3] := `Partial[ f[ g[ h[x]^2 ] ], x[1] ]`

Out[3] = $2\,\texttt{Gradient}[\mathrm{f}][\mathrm{g}[\mathrm{h}[\mathrm{x}]^2]].\mathrm{g}'[\mathrm{h}[\mathrm{x}]^2]\,\mathrm{h}[\mathrm{x}]\,\texttt{Partial}_1[\mathrm{h}][\mathrm{x}]$

Note that the answer above involves the dot product of two vectors—the gradient of f, evaluated at $g(h(x)^2)$, and the derivative of g, evaluated at $h(x)^2$. *Out[1]* should be compared with *Out[3]*.

Volume

`Volume[n]` is the volume (unnormalized) of the unit ball in $\mathbf{R}^n$:

In[1] := `Volume[4]`

Out[1] = $\dfrac{\texttt{Pi}^2}{2}$

`Volume` is computed using the formula given by Exercise 6 in Appendix A of [ABR].

SurfaceArea

`SurfaceArea[n]` is the surface area (unnormalized) of the unit sphere in $\mathbf{R}^n$:

In[1] := `SurfaceArea[61]`

$$Out[1] = \frac{2147483648\,\text{Pi}^{30}}{2921560637147316928501806009124925929687 5}$$

`SurfaceArea[n]` is computed by multiplying the formula for `Volume[n]` by n; see A.2 in Appendix A of [ABR].

IntegrateSphere

`IntegrateSphere[f, x]` equals $\int_S f\,d\sigma$. The integration with respect to normalized surface area measure $d\sigma$ takes place over the unit sphere S in the Euclidean space defined by x. Here f should be a polynomial function of x:

In[1] := `SetDimension[x, n]`

In[2] := `IntegrateSphere[ x[1]^2 x[2]^4 x[3]^6, x ]`

$$Out[2] = \frac{45}{\text{n}\,(2+\text{n})\,(4+\text{n})\,(6+\text{n})\,(8+\text{n})\,(10+\text{n})}$$

`IntegrateSphere` is computed by using the results in Section 3 of [W].

IntegrateBall

`IntegrateBall[f, x]` equals $\int_B f\,dV$. The integration with respect to (unnormalized) volume measure dV takes place over the unit ball B in the Euclidean space defined by x. Here f should be a function of x and $|x|$ that is a polynomial in x and a function in $|x|$ for which *Mathematica* can find an explicit antiderivative:

In[1] := `SetDimension[x, 7]`

In[2] := `IntegrateBall[ x[1]^2 x[2]^4 / (1 + Norm[x]), x ]`

$$Out[2] = \frac{2\,\text{Pi}^3\,(-18107+27720\,\text{Log}[2])}{12006225}$$

`IntegrateBall` is computed by converting to polar coordinates and then using the function `IntegrateSphere`.

Boundary Value Problems

HarmonicDecomposition

`HarmonicDecomposition[p, x]` gives the decomposition of p into a sum of harmonic polynomials (on the Euclidean space defined by x) times powers of $|x|$. Here p must be a polynomial function of x:

In[1] := `SetDimension[x, n]`

In[2] := `HarmonicDecomposition[ x[1]^4, x ]`

Out[2] = $\{\{\frac{3\,|x|^4}{(2+n)\,(4+n)} - \frac{6\,|x|^2\,x_1{}^2}{4+n} + x_1{}^4, 0\}, \{\frac{6\,(-|x|^2 + n\,x_1{}^2)}{n\,(4+n)}, 2\}, \{\frac{3}{n\,(2+n)}, 4\}\}$

The output given by `HarmonicDecomposition[p, x]` consists of a list of pairs. The first entry in each pair is a harmonic function, the second entry is the power of $|x|$ by which the first entry should be multiplied so that the sum of the resulting terms equals p. For example, in *Out[2]* we see that $\frac{3|x|^4}{(2+n)(4+n)} - \frac{6|x|^2 x_1{}^2}{4+n} + x_1{}^4$ and $\frac{6(-|x|^2+nx_1{}^2)}{n(4+n)}$ and $\frac{3}{n(2+n)}$ are harmonic functions of x; furthermore

$$x_1{}^4 = \frac{3|x|^4}{(2+n)(4+n)} - \frac{6|x|^2x_1{}^2}{4+n} + x_1{}^4 + \frac{6(-|x|^2+nx_1{}^2)}{n(4+n)}|x|^2 + \frac{3}{n(2+n)}|x|^4.$$

For a proof of the existence and uniqueness of the harmonic decomposition, see [ABR], Theorem 5.5 or [AR], Corollary 1.8. `HarmonicDecomposition` is computed using the algorithm described in Section 2 of [AR] (this paper is available electronically by e-mail, *gopher*, or anonymous *ftp*; see page 1 of this document for the appropriate electronic addresses).

AntiLaplacian

`AntiLaplacian[u, x]` gives an anti-Laplacian of u with respect to x; this is a function whose Laplacian with respect to x equals u. Here u must be a polynomial function of x or a sum of terms, each of which is a monomial in x times a function of $|x|$:

In[1] := `SetDimension[x, 6]`

In[2] := `AntiLaplacian[ x[1]^2 x[2] Norm[x]^3 Log[ Norm[x] ], x ]`

Out[2] = $\frac{1328\,|x|^7\,x_2 - 2730\,\mathrm{Log}[|x|]\,|x|^7\,x_2 - 33124\,|x|^5\,x_1{}^2\,x_2 + 124215\,\mathrm{Log}[|x|]\,|x|^5\,x_1{}^2\,x_2}{9316125}$

The function given by *Out[2]* has Laplacian equal to $x_1{}^2x_2|x|^3\log|x|$ on $\mathbf{R}^6$.

The anti-Laplacian of a given function is never unique. However, each polynomial has a unique anti-Laplacian that is a polynomial multiple of $|x|^2$, where x is the variable. Normally `AntiLaplacian` when applied to a polynomial uses an iterative algorithm that works rapidly but does not produce the anti-Laplacian that is a polynomial multiple of $|x|^2$. The option `Multiple -> Norm^2` will force `AntiLaplacian` to use an algorithm that is slower but produces the anti-Laplacian that is a polynomial multiple of $|x|^2$:

In[3] := `AntiLaplacian[ x[1]^2 x[2]^3, x, Multiple -> Norm^2 ]`

Out[3] = $$\frac{|x|^6 x_2 - 18\,|x|^4 x_1{}^2 x_2 - 6\,|x|^4 x_2{}^3 + 168\,|x|^2 x_1{}^2 x_2{}^3}{5376}$$

The function given by *Out[3]* is the unique polynomial multiple of $|x|^2$ whose Laplacian equals ${x_1}^2{x_2}^3$ on $\mathbf{R}^6$.

To understand how `AntiLaplacian` is computed, suppose we want to find an anti-Laplacian of $p(x)f(|x|)$, where p is a polynomial on $\mathbf{R}^n$ and f is a continuous function on $(0,\infty)$. Applying `HarmonicDecomposition` to p, using linearity, and replacing $f(|x|)$ with a power of $|x|$ times $f(|x|)$, we see that we only need to find anti-Laplacians of functions of the form $q(x)f(|x|)$, where q is a harmonic polynomial on $\mathbf{R}^n$ homogeneous of degree m. To find an anti-Laplacian of $q(x)f(|x|)$, suppose h is a twice-differentiable function on $(0,\infty)$. The Laplacian of $q(x)h(|x|)$ equals

$$q(x)\frac{(2m+n-1)h'(|x|)+|x|h''(|x|)}{|x|},$$

where we have used the product rule for Laplacians (see [ABR], page 13) and Exercise 20, Chapter 1 of [ABR]. We want to make the expression above equal $q(x)f(|x|)$, so we need only solve the differential equation

$$\frac{(2m+n-1)h'(t)+th''(t)}{t}=f(t).$$

A solution to this equation is given by

$$h(t)=\int_a^t s^{1-2m-n}\int_b^s r^{2m+n-1}f(r)\,dr\,ds$$

for any choice of the constants a and b. `AntiLaplacian` uses the formula above with $b=0$, although if a singularity is encountered at 0 it automatically chooses a different point (if you know there is a singularity at 0, you can save a bit of time by using the option `Singularity -> 0`).

Dirichlet

`Dirichlet[f, x]` is the solution to the standard Dirichlet problem: find the harmonic function on the unit ball in the Euclidean space defined by x that equals f on the unit sphere. Thus `Dirichlet[f, x]` is the Poisson integral of f as a function of x. Here f must be a polynomial function of x:

In[1] := `SetDimension[x, 5]`

In[2] := `Dirichlet[ x[1]^4 x[2], x ]`

Out[2] = $$\frac{11\,x_2 - 18\,|x|^2 x_2 + 7\,|x|^4 x_2 + 126\,x_1{}^2 x_2 - 126\,|x|^2 x_1{}^2 x_2 + 231\,x_1{}^4 x_2}{231}$$

The function given by *Out[2]* is harmonic on $\mathbf{R}^5$ and agrees with ${x_1}^4x_2$ on the unit sphere in $\mathbf{R}^5$.

`Dirichlet[f, g, x]`, with three arguments instead of two arguments as above, is the solution to the generalized Dirichlet problem: find the function on the unit ball of the Euclidean space defined by x that equals f on the unit sphere and whose Laplacian equals g. Here f must be polynomial function of x and g must be a function for which `AntiLaplacian` produces an explicit anti-Laplacian in closed form:

In[3] := `Dirichlet[ x[1]^2, Cos[ Norm[x] ], x ]`

$$Out[3] = \frac{1}{5} - 7\,\texttt{Cos}[1] - \frac{|\texttt{x}|^2}{5} + 4\,\texttt{Sin}[1] + \frac{8\,\texttt{Cos}[|\texttt{x}|]\,|\texttt{x}| - \texttt{Cos}[|\texttt{x}|]\,|\texttt{x}|^3 - 8\,\texttt{Sin}[|\texttt{x}|]\,|\texttt{x}| + 4\,|\texttt{x}|^2\,\texttt{Sin}[|\texttt{x}|]}{|\texttt{x}|^3} + \texttt{x}_1{}^2$$

The function given by *Out[3]* agrees with ${x_1}^2$ on the unit sphere in $\mathbf{R}^5$ and has Laplacian equal to $\cos|x|$. Although the term with $|x|^3$ in the denominator above looks as if it blows up at the origin, its value there is actually $1/3$.

`Dirichlet` with two arguments (the standard Dirichlet problem) is computed in standard fashion from `HarmonicDecomposition`; see Corollary 2.2 of [AR]. The solution to the generalized Dirichlet problem `Dirichlet[f, g, x]` equals h plus the solution to the standard Dirichlet problem with boundary data $f - h$, where h is an anti-Laplacian of g.

ExteriorDirichlet

`ExteriorDirichlet[f, x]` is the solution to the standard exterior Dirichlet problem: find the harmonic function on the exterior (including ∞) of the unit ball in the Euclidean space defined by x that equals f on the unit sphere. Thus `ExteriorDirichlet[f, x]` is the exterior Poisson integral of f as a function of x (see Theorem 4.8 of [ABR]). Here f must be a polynomial function of x:

In[1] := `SetDimension[x, 4]`

In[2] := `ExteriorDirichlet[ x[1]^4 x[2], x ]`

$$Out[2] = \frac{(3\,|\texttt{x}|^4 - 8\,|\texttt{x}|^6 + 5\,|\texttt{x}|^8 - 48\,|\texttt{x}|^2\,\texttt{x}_1{}^2 + 48\,|\texttt{x}|^4\,\texttt{x}_1{}^2 + 80\,\texttt{x}_1{}^4)\,\texttt{x}_2}{80\,|\texttt{x}|^{12}}$$

The function given by *Out[2]* is harmonic on $(\mathbf{R}^4 \cup \{\infty\}) \setminus \{0\}$ and agrees with ${x_1}^4 x_2$ on the unit sphere in $\mathbf{R}^4$.

`ExteriorDirichlet[f, g, x]`, with three arguments instead of two arguments as above, is a solution to the generalized exterior Dirichlet problem: find a function on the exterior of the unit ball of the Euclidean space defined by x that equals f on the unit sphere and whose Laplacian equals g. Here f must be polynomial function of x and g must be a function for which `AntiLaplacian` produces an explicit anti-Laplacian in closed form. A solution to the generalized exterior Dirichlet problem `ExteriorDirichlet[f, g, x]` equals h plus the solution to the standard exterior Dirichlet problem with boundary data $f - h$, where h is an anti-Laplacian of g. The generalized exterior Dirichlet problem does not have a unique solution—any solution can be perturbed by the addition of a harmonic function on the exterior of the unit ball that equals 0 on the unit sphere. The algorithm used here to find the anti-Laplacian seems to produce the solution that grows at ∞ as slowly as possible:

In[3] := `ExteriorDirichlet[ x[1]^3 x[2], Norm[x]^(-3), x ]`

$$Out[3] = \frac{8\,|x|^8 - 8\,|x|^9 - 3\,|x|^2\,x_1\,x_2 + 3\,|x|^4\,x_1\,x_2 + 8\,x_1{}^3\,x_2}{8\,|x|^{10}}$$

The function given by *Out[3]* has Laplacian equal to $|x|^{-3}$ and agrees with ${x_1}^3 x_2$ on the unit sphere in $\mathbf{R}^4$.

AnnularDirichlet

`AnnularDirichlet[f, g, r, s, x]` is the solution to the annular Dirichlet problem: find the harmonic function on the annular region with inner radius r and outer radius s that equals f on the sphere of radius r and equals g on the sphere of radius s. Here f and g must be polynomial functions of x:

In[1] := `SetDimension[x, n]`

In[2] := `AnnularDirichlet[ x[1]^2, x[1] x[2] x[3], 2, 3, x ]`

$$Out[2] = \frac{4\,(\frac{-9}{3^n} + |x|^{2-n})}{(\frac{4}{2^n} - \frac{9}{3^n})\,n} + \frac{(\frac{-1}{9\,3^n} + |x|^{-2-n})\,(-(\frac{|x|^2}{n}) + x_1{}^2)}{\frac{1}{4\,2^n} - \frac{1}{9\,3^n}} + \frac{(-2^{-4-n} + |x|^{-4-n})\,x_1\,x_2\,x_3}{-2^{-4-n} + 3^{-4-n}}$$

The function given by *Out[2]* is harmonic on $\mathbf{R}^n \setminus \{0\}$, agrees with ${x_1}^2$ on the sphere of radius 2, and agrees with $x_1 x_2 x_3$ on the sphere of radius 3.

`AnnularDirichlet[f, g, h, r, s, x]`, with six arguments instead of five arguments as above, is the solution to the generalized annular Dirichlet problem: find the function on the annular region with inner radius r and outer radius s that equals f on the sphere of radius r, equals g on the sphere of radius s, and has Laplacian equal to h. Here f and g must be polynomial functions of x and h must be a function for which `AntiLaplacian` produces an explicit anti-Laplacian in closed form:

In[3] := `SetDimension[x, 3]`

In[4] := `AnnularDirichlet[ x[3], x[2], x[1], 1, 5, x ]`

$$Out[4] = \frac{1500\,x_1 - 1562\,|x|^3\,x_1 + 62\,|x|^5\,x_1 - 625\,x_2 + 625\,|x|^3\,x_2 + 625\,x_3 - 5\,|x|^3\,x_3}{620\,|x|^3}$$

The function given by *Out[4]* equals x_3 on the sphere in $\mathbf{R}^3$ of radius 1, equals x_2 on the sphere of radius 5, and has Laplacian equal to x_1.

`AnnularDirichlet` with five arguments (the standard annular Dirichlet problem) is computed by using the ideas in the proof of Theorem 10.13 of [ABR]; also see Section 2 of [AR]. The solution to the generalized annular Dirichlet problem `AnnularDirichlet[f, g, h, r, s, x]` equals u plus the solution to the standard Dirichlet problem with boundary data $f - u, g - u$, where u is an anti-Laplacian of h.

Neumann

`Neumann[f, x]` is the solution to the standard Neumann problem: find the harmonic function on the unit ball in the Euclidean space defined by x whose outward normal derivative on the unit sphere equals f and whose value at the origin equals 0. Here f must be a polynomial function of x. Furthermore, the integral of f over the unit sphere with respect to surface area measure must equal 0 (Green's identity shows that this condition is necessary for the existence of a solution to the standard Neumann problem):

In[1] := `SetDimension[x, 4]`

In[2] := `Neumann[ x[1]^4 x[2], x ]`

$$Out[2] = \frac{x_2}{16} + \frac{-(|x|^2 x_2) + 6 x_1{}^2 x_2}{30} + \frac{3 |x|^4 x_2 - 48 |x|^2 x_1{}^2 x_2 + 80 x_1{}^4 x_2}{400}$$

The function given by *Out[2]* is harmonic on $\mathbf{R}^4$ and has an outward normal derivative on the unit sphere that equals ${x_1}^4 x_2$.

`Neumann[f, g, x]`, with three arguments instead of two arguments as above, is the solution to the generalized Neumann problem: find the function on the unit ball of the Euclidean space defined by x whose outward normal derivative on the unit sphere equals f, whose Laplacian equals g, and whose value at the origin equals 0. Here f must be a polynomial function of x and g must be a function for which `AntiLaplacian` produces an explicit anti-Laplacian in closed form. Furthermore, the integral of f over the unit sphere with respect to (unnormalized) surface area measure must equal the integral of g over the unit ball with respect to (unnormalized) volume measure (Green's identity shows that this condition is necessary for the existence of a solution to the generalized Neumann problem):

In[3] := `Neumann[ x[1]^3, x[2]^3 Norm[x]^2, x ]`

$$Out[3] = \frac{720 x_1 - 240 |x|^2 x_1 + 480 x_1{}^3 - 84 x_2 + 35 |x|^2 x_2 - 3 |x|^6 x_2 - 70 x_2{}^3 + 30 |x|^4 x_2{}^3}{1440}$$

The function given by *Out[3]* has an outward normal derivative on the unit sphere in $\mathbf{R}^4$ that equals ${x_1}^3$ and has Laplacian equal to ${x_2}^3|x|^2$.

The solution to the standard Neumann problem is computed from `HarmonicDecomposition`; see Section 2 of [AR]. The solution to the generalized Neumann problem `Neumann[f, g, x]` equals u plus the solution to the standard Neumann problem with boundary data $f - h$, where u is an anti-Laplacian of g with $u(0) = 0$ and h is the outward normal derivative of u on the unit sphere.

ExteriorNeumann

`ExteriorNeumann[f, x]` is the solution to the standard exterior Neumann problem: find the harmonic function on the exterior of the unit ball in the Euclidean space defined by x whose outward normal derivative on the unit sphere equals f and whose limit at ∞ equals 0. Here f must be a polynomial function of x. If the dimension of the Euclidean space equals 2, then the integral of f over the unit circle with respect to arc length measure must equal 0:

In[1] := `SetDimension[x, 4]`

In[2] := `ExteriorNeumann[ x[1]^4 x[2], x ]`

$$Out[2] = \frac{x_2}{48\,|x|^4} + \frac{-(|x|^2\,x_2) + 6\,x_1{}^2\,x_2}{50\,|x|^8} + \frac{3\,|x|^4\,x_2 - 48\,|x|^2\,x_1{}^2\,x_2 + 80\,x_1{}^4\,x_2}{560\,|x|^{12}}$$

The function given by *Out[2]* is harmonic on $\mathbf{R}^4 \setminus \{0\}$, has limit 0 at ∞, and has an outward normal derivative (with respect to the exterior of the unit ball) on the unit sphere that equals ${x_1}^4 x_2$.

`ExteriorNeumann[f, g, x]`, with three arguments instead of two arguments as above, is the solution to the generalized exterior Neumann problem: find the function on the exterior of the unit ball of the Euclidean space defined by x whose outward normal derivative on the unit sphere equals f and whose Laplacian equals g. Here f must be a polynomial function of x and g must be a function for which `AntiLaplacian` produces an explicit anti-Laplacian in closed form. The generalized exterior Neumann problem does not have a unique solution—any solution can be perturbed by the addition of a constant. If the dimension of the Euclidean space equals 2, then the integral of f over the unit circle with respect to arc length measure must equal the integral of $-g$ over the unit disk with respect to area measure:

In[3] := `ExteriorNeumann[ x[1]^2 x[2], Norm[x]^(-4), x ]`

$$Out[3] = \frac{-45\,|x|^6 - 90\,\mathrm{Log}[|x|]\,|x|^6 - 6\,|x|^2\,x_2 + 10\,|x|^4\,x_2 + 36\,x_1{}^2 x_2}{180\,|x|^8}$$

The function given by *Out[3]* has an outward normal derivative (with respect to the exterior of the unit ball) on the unit sphere in $\mathbf{R}^4$ that equals ${x_1}^2 x_2$ and has Laplacian equal to $|x|^{-4}$.

BiDirichlet

`BiDirichlet[f, x]` is the solution to the standard biDirichlet problem: find the biharmonic function on the unit ball in the Euclidean space defined by x that equals f on the unit sphere and whose normal derivative on the unit sphere equals 0. (A function is called biharmonic if the Laplacian of its Laplacian equals 0.) Here f must be a polynomial function of x:

In[1] := `SetDimension[x, 3]`

In[2] := `BiDirichlet[ x[1]^3 x[2], x ]`

$$Out[2] = \frac{6\,x_1\,x_2 - 12\,|x|^2\,x_1\,x_2 + 6\,|x|^4\,x_1\,x_2 + 21\,x_1{}^3\,x_2 - 14\,|x|^2\,x_1{}^3\,x_2}{7}$$

The function given by *Out[2]* is biharmonic on $\mathbf{R}^3$, agrees with ${x_1}^3 x_2$ on the unit sphere, and has a normal derivative on the unit sphere equal to 0.

`BiDirichlet[f, g, x]`, with three arguments instead of two arguments as above, is the solution to the generalized BiDirichlet problem: find the biharmonic function on the unit ball of the Euclidean space defined by x that equals f on the unit sphere and whose outward normal derivative on the unit sphere equals g. Here f and g must be polynomial functions of x:

In[3] := `BiDirichlet[ x[1]^2, x[2]^2 x[3], x ]`

Out[3] = $$\frac{1 - 2\,|x|^2 + |x|^4 + 6\,x_1{}^2 - 3\,|x|^2\,x_1{}^2}{3} + \frac{(-1 + |x|^2)\,(x_3 - |x|^2\,x_3 + 5\,x_2{}^2\,x_3)}{10}$$

The function given by *Out[3]* is biharmonic on $\mathbf{R}^3$, agrees with ${x_1}^2$ on the unit sphere, and has an outward normal derivative on the unit sphere equal to ${x_2}^2 x_3$.

Spherical Harmonics

BasisH

`BasisH[m, x]` gives a vector space basis for $\mathcal{H}_m(\mathbf{R}^n)$, the space of homogeneous harmonic polynomials on $\mathbf{R}^n$ of degree m, where n is the dimension of x. The user must first set the dimension of x to an integer value. We recommend wrapping the *Mathematica* command `TableForm` around `BasisH[m, x]` to produce an output display that is more readable than the usual list format:

In[1] := `SetDimension[x, 3]`

In[2] := `TableForm[ BasisH[4, x] ]`

Out[2] =

$$3\,|x|^4 - 30\,|x|^2\,x_2{}^2 + 35\,x_2{}^4$$
$$3\,|x|^2\,x_2\,x_3 - 7\,x_2{}^3\,x_3$$
$$|x|^4 - 5\,|x|^2\,x_2{}^2 - 5\,|x|^2\,x_3{}^2 + 35\,x_2{}^2\,x_3{}^2$$
$$3\,|x|^2\,x_2\,x_3 - 7\,x_2\,x_3{}^3$$
$$3\,|x|^4 - 30\,|x|^2\,x_3{}^2 + 35\,x_3{}^4$$
$$3\,|x|^2\,x_1 x_2 - 7\,x_1\,x_2{}^3$$
$$|x|^2\,x_1\,x_3 - 7\,x_1\,x_2{}^2\,x_3$$
$$|x|^2\,x_1\,x_2 - 7\,x_1\,x_2\,x_3{}^2$$
$$3\,|x|^2\,x_1\,x_3 - 7\,x_1\,x_3{}^3$$

The option `Orthonormal -> Ball`, when used with `BasisH`, will produce a basis that is orthonormal with respect to the inner product on $L^2(B, dV)$ (B denotes the open unit ball in Euclidean space):

In[3] := `TableForm[ BasisH[ 2, x, Orthonormal -> Ball ] ]`

Out[3] =

$$\frac{\text{Sqrt}[\frac{35}{\text{Pi}}]\,(-|\text{x}|^2 + 3\,\text{x}_2{}^2)}{4}$$

$$\frac{\text{Sqrt}[\frac{105}{\text{Pi}}]\,\text{x}_2\,\text{x}_3}{2}$$

$$\frac{\text{Sqrt}[\frac{105}{\text{Pi}}]\,(-|\text{x}|^2 + \text{x}_2{}^2 + 2\,\text{x}_3{}^2)}{4}$$

$$\frac{\text{Sqrt}[\frac{105}{\text{Pi}}]\,\text{x}_1\,\text{x}_2}{2}$$

$$\frac{\text{Sqrt}[\frac{105}{\text{Pi}}]\,\text{x}_1\,\text{x}_3}{2}$$

The last output is a basis for $\mathcal{H}_2(\mathbf{R}^3)$ that is orthonormal with respect to the inner product on $L^2(B, dV)$. The Euclidean space has dimension 3 here because x was defined to have dimension 3 in *In[1]*.

To produce a basis for $\mathcal{H}_m(S)$, the space of spherical harmonics on the unit sphere S, the user should first set $|x|$ equal to 1. In addition, the option `Orthonormal -> Sphere`, when used with `BasisH`, will produce a basis that is orthonormal with respect to the inner product on $L^2(S, d\sigma)$:

In[4] := `Norm[x] = 1 ;`

In[5] := `TableForm[ BasisH[ 4, x, Orthonormal -> Sphere ] ]`

Out[5] =

$$\frac{9 - 90\,\text{x}_2{}^2 + 105\,\text{x}_2{}^4}{8}$$

$$\frac{-45\,\text{x}_2\,\text{x}_3 + 105\,\text{x}_2{}^3\,\text{x}_3}{2\,\text{Sqrt}[10]}$$

$$\frac{3\,\text{Sqrt}[5]\,(1 - 8\,\text{x}_2{}^2 + 7\,\text{x}_2{}^4 - 2\,\text{x}_3{}^2 + 14\,\text{x}_2{}^2\,\text{x}_3{}^2)}{4}$$

$$\frac{3\,\text{Sqrt}[\frac{35}{2}]\,(-3\,\text{x}_2\,\text{x}_3 + 3\,\text{x}_2{}^3\,\text{x}_3 + 4\,\text{x}_2\,\text{x}_3{}^3)}{2}$$

$$\frac{3\,\text{Sqrt}[35]\,(1 - 2\,\text{x}_2{}^2 + \text{x}_2{}^4 - 8\,\text{x}_3{}^2 + 8\,\text{x}_2{}^2\,\text{x}_3{}^2 + 8\,\text{x}_3{}^4)}{8}$$

$$\frac{-45\,\text{x}_1\,\text{x}_2 + 105\,\text{x}_1\,\text{x}_2{}^3}{2\,\text{Sqrt}[10]}$$

$$\frac{3\,\text{Sqrt}[5]\,(-(\text{x}_1\,\text{x}_3) + 7\,\text{x}_1\,\text{x}_2{}^2\,\text{x}_3)}{2}$$

$$\frac{3\,\text{Sqrt}[\frac{35}{2}]\,(-(\text{x}_1\,\text{x}_2) + \text{x}_1\,\text{x}_2{}^3 + 4\,\text{x}_1\,\text{x}_2\,\text{x}_3{}^2)}{2}$$

$$\frac{3\,\text{Sqrt}[35]\,(-(\text{x}_1\,\text{x}_3) + \text{x}_1\,\text{x}_2{}^2\,\text{x}_3 + 2\,\text{x}_1\,\text{x}_3{}^3)}{2}$$

`BasisH` is computed by using Theorem 5.34 of [ABR].

ZonalHarmonic

`ZonalHarmonic[m, x, y]` is the extended zonal harmonic $Z_m(x,y)$ as defined by equation 8.7 of [ABR]:

In[1] := `SetDimension[x, n]`

In[2] := `ZonalHarmonic[3, x, y]`

$$Out[2] = \frac{n\,(2+n)\,(4+n)\,(x.y)^3}{6} - \frac{n\,(4+n)\,x.y\,|x|^2\,|y|^2}{2}$$

To obtain the zonal harmonic $Z_m(x,z)$ as a homogeneous polynomial in x of degree m with pole z on the unit sphere (as in Theorem 5.24 of [ABR]), the user should tell the computer that $|z| = 1$:

In[3] := `SetDimension[x, 6]`

In[4] := `Norm[z] = 1 ;`

In[5] := `ZonalHarmonic[4, x, z]`

$$Out[5] = 240\,(x.z)^4 - 144\,(x.z)^2\,|x|^2 + 9\,|x|^4$$

To obtain the zonal harmonic $Z_m(x,z)$ as an element of $\mathcal{H}_m(S)$ as originally defined in Chapter 5 of [ABR], the user should tell the computer that $|x| = 1$ and $|z| = 1$.
`ZonalHarmonic` is computed by using Theorem 5.24 of [ABR].

DimensionH

`DimensionH[m, n]` is the vector space dimension of $\mathcal{H}_m(\mathbf{R}^n)$, the space of harmonic polynomials on $\mathbf{R}^n$ homogeneous of degree m:

In[1] := `DimensionH[12, 100]`

Out[1] = `3901030682812965`

`DimensionH` is computed by using equation 5.17 of [ABR].

Inversion and the Kelvin Transform

Inversion

`Inversion[x]` is the inversion of x, as defined in Chapter 4 of [ABR], where the inversion of

x was denoted by x^*:

In[1] := `Inversion[ 2x + y ]`

Out[1] = $|2\,x + y|^{-2}\,(2\,x + y)$

Suppose E is the sphere with center a, radius r and x is a point in $\mathbf{R}^n$. Exercise 11 in Chapter 4 of [ABR] asks for a proof that x^* and $(x_E)^*$ are symmetric about E^*, which is the same as showing that $(x^*)_{E^*} = (x_E)^*$. Using Exercise 10 in Chapter 4 of [ABR], along with equation 4.10 from the same book, we see that the equation $(x^*)_{E^*} = (x_E)^*$ is equivalent to the following equation:

$$\frac{a}{|a|^2 - r^2} + \frac{r^2}{(r^2 - |a|^2)^2}(x^* - \frac{a}{|a|^2 - r^2})^* = (a + r^2(x-a)^*)^*.$$

Let's see how we can verify the equation above by computer. We will subtract the left hand side from the right hand side, hoping to get 0:

In[2] :=
```
Inversion[ a + r^2 Inversion[x - a] ] -
 a/(Norm[a]^2 - r^2) -
 r^2/(r^2 - Norm[a]^2)^2 Inversion[
 Inversion[x] - a/(Norm[a]^2 - r^2) ]
```

Out[2] = $\frac{1}{r^2 - |a|^2}\,a - \frac{r^2}{(r^2 - |a|^2)^2\,|\frac{1}{r^2 - |a|^2}\,a + |x|^{-2}\,x|^2}\,(\frac{1}{r^2 - |a|^2}\,a + |x|^{-2}\,x) +$

$|a + \frac{r^2}{|-a + x|^2}\,(-a + x)|^{-2}\,(a + \frac{r^2}{|-a + x|^2}\,(-a + x))$

The last output doesn't look much like 0, so let's apply `ExpandNorm` to it:

In[3] := `ExpandNorm[%]`

Out[3] = `0`

It worked! Thus we have proved the assertion of Exercise 11 in Chapter 4 of [ABR].

Kelvin

`Kelvin[u, x]` is the Kelvin transform of u, thought of as a function of x:

In[1] := `SetDimension[x, n]`

In[2] := `Kelvin [ (x.y)^2 + x[3]^5 Norm[x], x ]`

Out[2] = $|x|^{-9-n}\,((x.y)^2\,|x|^7 + x_3{}^5)$

Phi

`Phi[z]` is is the modified inversion $\Phi(z)$ introduced in our study of half-spaces in Chapter 7 of [ABR]. Here is a proof of the assertion made there that $\Phi\big(\Phi(z)\big) = z$:

In[1] := `Phi[z]`

$$Out[1] = \mathtt{S} + \frac{2}{|-\mathtt{S}+\mathtt{z}|^2}(-\mathtt{S}+\mathtt{z})$$

In[2] := `Phi[%]`

Out[2] = `z`

In *Out[1]*, `S` denotes the south pole **S**, as in Chapter 7 of [ABR].

Another, and often more convenient, form of `Phi` is given by `Phi[x, y]`. Here x is a vector, y is a real number, and thus (x, y) is a vector in a Euclidean space whose dimension is one more than the dimension of x. Here is a proof of the assertion of Exercise 6 in Chapter 7 of [ABR] that $1 - |\Phi(x,y)|^2 = 4y/\big(|x|^2 + (y+1)^2\big)$:

In[3] := `Phi[x, y]`

$$Out[3] = \{\frac{2}{(1+\mathtt{y})^2 + |\mathtt{x}|^2}\,\mathtt{x},\ -1 + \frac{2\,(1+\mathtt{y})}{(1+\mathtt{y})^2 + |\mathtt{x}|^2}\}$$

In[4] := `1 - Norm[%]^2`

$$Out[4] = \frac{4\,\mathtt{y}}{1 + 2\,\mathtt{y} + \mathtt{y}^2 + |\mathtt{x}|^2}$$

KelvinM

`KelvinM[u, z]` is the modified Kelvin transform $\mathcal{K}[u](z)$ introduced in our study of half-spaces in Chapter 7 of [ABR]. Here is a proof of the assertion made there that $\mathcal{K}[\mathcal{K}[u]] = u$:

In[1] := `SetDimension[z, n]`

In[2] := `KelvinM[ u[z], z ]`

$$Out[2] = \frac{2^{\mathtt{n}/2}\,|-\mathtt{S}+\mathtt{z}|^{2-\mathtt{n}}\,\mathtt{u}[\mathtt{S} + \frac{2}{|-\mathtt{S}+\mathtt{z}|^2}(-\mathtt{S}+\mathtt{z})]}{2}$$

In[3] := `KelvinM[%, z]`

Out[3] = `u[z]`

Another form of `KelvinM` is given by `KelvinM[u, x, y]`. Here x is a vector, y is a real variable, and u is a function of (x, y), which is a vector in a Euclidean space whose dimension is one more than the dimension of x:

In[4] := `SetDimension[ x, n-1 ]`

In[5] := `KelvinM[ x[4], x, y ]`

$$Out[5] = \frac{2^{n/2}\, x_4}{((1+y)^2 + |x|^2)^{n/2}}$$

If we had used the expression `KelvinM[ z[4], z ]`, the result would have included terms of the form `S[4]`, which denotes the fourth coordinate of the south pole **S**. Of course `S[4]` equals either 0 or -1, but since the dimension of z equals the symbol n (from *In[1]*), the computer has no way of determining whether n equals 4. With the command `KelvinM[ x[4], x, y ]` there is no problem; x_4 cannot be the last coordinate, because the last coordinate is denoted by y.

Kernels

PoissonKernel

`PoissonKernel[x, y]` is the extended Poisson kernel $P(x, y)$ for the unit ball as defined by equation 8.11 of [ABR]:

In[1] := `SetDimension[x, n]`

In[2] := `PoissonKernel[x, y]`

$$Out[2] = \frac{1 - |x|^2\, |y|^2}{(1 - 2\, x.y + |x|^2\, |y|^2)^{n/2}}$$

The usual Poisson kernel $P(x, z)$, where z lies on the unit sphere, can be obtained by telling the computer that $|z| = 1$:

In[3] := `Norm[z] = 1 ;`

In[4] := `PoissonKernel[x, z]`

$$Out[4] = \frac{1 - |x|^2}{(1 - 2\, x.z + |x|^2)^{n/2}}$$

PoissonKernelH

`PoissonKernelH[z, w]` is the extended Poisson kernel $P_H(z, w)$ for the upper half-space H, as defined by equation 8.20 of [ABR]:

In[1] := `SetDimension[z, 5]`

In[2] := `PoissonKernelH[z, w]`

Out[2] = $$\frac{3\,(\mathtt{w}_5 + \mathtt{z}_5)}{4\,\mathtt{Pi}^2\,(-2\,\mathtt{w.z} + |\mathtt{w}|^2 + |\mathtt{z}|^2 + 4\,\mathtt{w}_5\,\mathtt{z}_5)^{5/2}}$$

The usual Poisson kernel $P_H(z, w)$ for the upper half-space, where we think of $w \in \partial H$, can be obtained by telling the computer that the last coordinate of w is 0. In the last example this could be done by first typing `w[5] = 0`.

`PoissonKernelH[x, y, t, u]` equals $P_H\big((x, y), (t, u)\big)$ and thus is another format for the extended Poisson kernel on the upper half-space. Here x and t denote vectors, and y and u denote nonnegative numbers. Note that if we want to work in the upper half-space of $\mathbf{R}^n$, then the dimension of x should be set to $n - 1$:

In[5] := `SetDimension[ x, n-1 ]`

In[6] := `PoissonKernelH[x, y, t, u]`

Out[6] = $$\frac{2\,(\mathtt{u} + \mathtt{y})}{\mathtt{n}\,((\mathtt{u} + \mathtt{y})^2 + |-\mathtt{t} + \mathtt{x}|^2)^{\mathtt{n}/2}\,\mathtt{Volume}[\mathtt{n}]}$$

The usual Poisson kernel $P_H\big((x, y), t\big)$ for the upper half-space, where $t \in \partial H$, can be obtained by typing `PoissonKernelH[x, y, t, 0]`.

BergmanKernel

`BergmanKernel[x, y]` is the reproducing kernel for the harmonic Bergman space of the unit ball:

In[1] := `SetDimension[x, 10]`

In[2] := `BergmanKernel[x, y]`

Out[3] = $$\frac{12\,(10 + 8\,(-3 + \mathtt{x.y})\,|\mathtt{x}|^2\,|\mathtt{y}|^2 + 6\,|\mathtt{x}|^4\,|\mathtt{y}|^4)}{\mathtt{Pi}^5\,(1 - 2\,\mathtt{x.y} + |\mathtt{x}|^2\,|\mathtt{y}|^2)^6}$$

`BergmanKernel` is computed by using Theorem 8.13 of [ABR].

BergmanKernelH

`BergmanKernelH[z, w]` is the reproducing kernel for the harmonic Bergman space of the upper half-space:

In[1] := `SetDimension[z, 10]`

In[2] := `BergmanKernelH[z, w]`

$$Out[2] = \frac{48\,(2\,w.z - |w|^2 - |z|^2 + 16\,w_{10}\,z_{10} + 10\,(w_{10}{}^2 + z_{10}{}^2))}{Pi^5\,(2\,w.z - |w|^2 - |z|^2 - 4\,w_{10}\,z_{10})^6}$$

`BergmanKernelH[x, y, t, u]` is another format for the reproducing kernel for the harmonic Bergman space of the upper half-space. Here x and t denote vectors, y and u denote nonnegative numbers, and the kernel is evaluated at $\big((x, y), (t, u)\big)$. Note that if we want to work in the upper half-space of $\mathbf{R}^n$, then the dimension of x should be set to $n - 1$:

In[3] := `SetDimension[ x, n-1 ]`

In[4] := `BergmanKernelH[x, y, t, u]`

$$Out[4] = \frac{4\,((-1 + n)\,(u + y)^2 - |-t + x|^2)\,((u + y)^2 + |-t + x|^2)^{-1-n/2}}{n\,Volume[n]}$$

`BergmanKernelH` is computed by using Theorem 8.22 of [ABR].

Miscellaneous

BergmanProjection

`BergmanProjection[u, x]` is the orthogonal projection of u onto $b^2(B)$, the harmonic Bergman space of the unit ball (see Chapter 8 of [ABR]). Here u must be a polynomial function of x:

In[1] := `SetDimension[x, 5]`

In[2] := `BergmanProjection[ x[1]^2 x[2]^3, x ]`

$$Out[2] = \frac{x_2}{33} + \frac{|x|^4\,x_2}{33} - \frac{3\,|x|^3\,x_1{}^2\,x_2}{11} - \frac{|x|^2\,x_2{}^3}{11} + x_1{}^2\,x_2{}^3 + \frac{-6\,|x|^2\,x_2 + 21\,x_1{}^2\,x_2 + 7\,x_2{}^3}{91}$$

`BergmanProjection` is computed by using Theorem 8.14 of [ABR].

Explicit Coordinates

Throughout these notes we have worked with symbolic vectors, usually called x or z. Vectors can also be described by giving explicit coordinates in the form of a list. Both formats (symbolic vectors and explicit lists) work with all functions in our package. For example, suppose we are working in $\mathbf{R}^3$ and want to use x, y, and z instead of x_1, x_2, and x_3. Here is how to find the Poisson integral of x^2yz:

In[1] := `Dirichlet[ x^2 y z, {x, y, z} ]`

Out[1] = $\dfrac{y z + 6 x^2 y z - y^3 z - y z^3}{7}$

Explicit coordinates, when used as arguments of non-explicit functions, must be enclosed in set brackets. Thus to find the Laplacian of the function whose value at a point $(x, y, z) \in \mathbf{R}^3$ is $f(x^5, 5x^2 y, x^3 z^4)$, use `Laplacian[ f[ {x^5, 5x^2 y, x^3 z^4}], {x, y, z} ]`. Try it!

HarmonicConjugate

`HarmonicConjugate[u, x, y]` is the harmonic conjugate of u on $\mathbf{R}^2$, where the coordinates in $\mathbf{R}^2$ are denoted by x, y and u is a harmonic function of (x, y):

In[1] := `u = 15x^2 y + 12x^3 y - 5y^3 - 12x y^3 ;`

In[2] := `HarmonicConjugate[u, x, y]`

Out[2] = $-5 x^3 - 3 x^4 + 3 x (5 + 6x) y^2 - 3 y^4$

`HarmonicConjugate` is computed by using Exercise 6 in Chapter 1 of [ABR].

Schwarz

`Schwarz[x]` is the maximum of $|u(x)|$, where u ranges over all harmonic functions on the unit ball (in the Euclidean space whose dimension is determined by x) with $u(0) = 0$ and $|u| < 1$. Thus, in the notation of Theorem 6.16 of [ABR], `Schwarz[x]` equals $U(|x|\mathbf{N})$. The user must first set the dimension of the Euclidean space to an integer value:

In[1] := `SetDimension[x, 5]`

In[2] := `Schwarz[x]`

Out[2] = $|x|^{-3} + \dfrac{(-1 + |x|)(1 + |x|)(2 + |x|^2)(1 + 2|x|^2)}{2|x|^3 (1 + |x|^2)^{3/2}}$

`Schwarz[x]` is computed by evaluating the Poisson integral that defines $U(|x|\mathbf{N})$, iterating an appropriate formula from Appendix A of [ABR]. The integrals that *Mathematica* must evaluate in this process are complicated; be prepared for your computer to use nontrivial amounts of time and memory.

TurnOff and TurnOn

On rare occasions the user may want to turn off one or more of the features built into our software package. `TurnOff[F]` turns off feature `F`, where `F` equals one of the features described below (`Togetherness` or `Subscripts`). `TurnOn[F]` turns feature `F` back on. All features described below are automatically turned on when the software package is initially loaded into the computer, so most users will never need to use these commands.

Togetherness

When not using our software package, *Mathematica* might tell the user that the result of a calculation is $t^2 + (1 - t)(1 + t)$. Of course the user will easily recognize that the last quantity equals 1, but with more complicated results the user will need to apply appropriate *Mathematica* commands (such as `Simplify`, `Expand`, or `Together`) to get the result in simplest form. Applying `Simplify` to all *Mathematica* output is a bad idea, because *Mathematica* can take a long time to realize that some expressions cannot be simplified. We have found that for almost all output generated by our package, the result is already in its simplest form or applying `Together` to it and all its subexpressions puts it in simplest form. Thus we have redefined the *Mathematica* variable `$Post` in our software package so that for each output `X`, *Mathematica* will display either `X` or the result of applying `Together` to `X` and all its subexpressions, whichever is shorter (if the user had defined `$Post` before starting our package, our version of `$Post` is composed with the user's). This procedure is invisible to the user and requires only a short amount of computer time. If you want to turn off this feature, use the command `TurnOff[ Togetherness ]` (if the user had defined `$Post` before starting our package, this restores the user's version). The command `TurnOn[ Togetherness ]` turns this feature back on again.

If `Homogeneous` or `Taylor` is used to find an expansion about a point other than the origin, then `Togetherness` will be automatically turned off. If this were not done then, for example, *Out[2]* in the *Taylor* subsection earlier in these notes would be simplified to $1+x_1x_2+{x_1}^2$, which is not what the user wanted to see when asking for an expansion about b.

Subscripts

When using our software package, the j^{th} coordinate of a vector x will be output using the standard mathematical notation $\mathtt{x}_\mathtt{j}$, and the partial derivative of a function f with respect to the j^{th} coordinate will be denoted by $\mathtt{Partial}_\mathtt{j}[\mathtt{f}]$. The command `TurnOff[ Subscripts ]` turns off these features, so that the output will appear as `x[j]` and `Partial[{j}][f]`. The command `TurnOn[ Subscripts ]` turns these features back on.

References

[ABR] Sheldon Axler, Paul Bourdon, and Wade Ramey, *Harmonic Function Theory*, Graduate Texts in Mathematics, Springer-Verlag, 1992.

[AR] Sheldon Axler and Wade Ramey, Harmonic polynomials and Dirichlet-type problems, *Proceedings of the American Mathematical Society* 123 (1995), 3765–3773.

[W] Hermann Weyl, On the volume of tubes, *American Journal of Mathematics* 61 (1939), 461–472.

Index