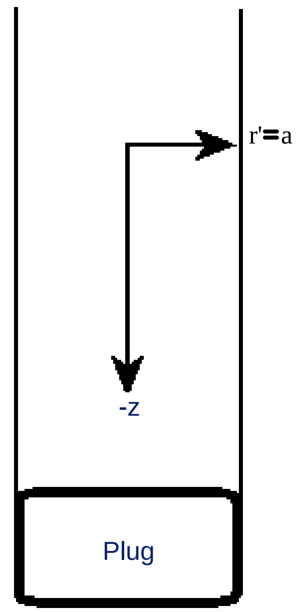
Separation of Variables. Application of eigenvalue-eigenvector to solution of PDE. Start-up flow in a pipe.
Instructor: Nam Sun Wang

Problem Statement. A fluid of viscosity μ and density ρ fills the inside of a very long vertical cylindrical pipe of radius a. We suddenly remove the plug initially placed at the bottom of the pipe at time t'=0. Find the velocity profile. The only driving force is gravity.



The radial and angular components of the velocity field are 0, and only velocity component is in z-direction. v_z' is a function of both time t' and radius r'.

$$v_r = v_\theta = 0 \qquad v_{z'} = v_{z'}(t', r')$$

Step 0. Develop a mathematical model. The following partial differential equation (PDE) describes the development of velocity in the z-direction. Note that the differential equation describes the laws of nature|physics -- the same laws of physics apply to all problems and never change; they remain the same beyond the day we die. Whereas, the boundary conditions are problem-specific (man-made) and depends on the particular physical set up or initial condition.

$$\rho \cdot \frac{d}{dt'} v_{z'} = -\rho \cdot g \cdot h + \frac{\mu}{r'} \cdot \frac{d}{dr'}\left(r' \cdot \frac{d}{dr'} v_{z'}\right)$$

B.C. at $t' = 0$ $v_z = 0$
at $r' = a$ $v_z = 0$

Step 1. Non-dimensionalization. We usually scale everything to O(1).

$$r = \frac{r'}{a} \qquad t = \frac{t'}{\dfrac{a^2 \cdot \rho}{\mu}} \qquad v_z = \frac{v_{z'}}{\dfrac{\rho \cdot g \cdot h \cdot a^2}{4 \cdot \mu}}$$

$$\frac{d}{dt} v_z = -4 + \frac{1}{r} \cdot \frac{d}{dr}\left(r \cdot \frac{d}{dr} v_z\right)$$

B.C. at $t = 0$ $v_{z0}(r) := 0$
at $r = 1$ $v_z = 0$

Step 2. Find steady-state solution. (See side note at the end) $v_{ss}(r) := r^2 - 1$

Step 3. Work in deviation variable w $w = v_z - v_{ss} = v_z + \left(1 - r^2\right)$

This is mathematically equivalent to eliminating the constant term "-4" by shifting the dependent variable v_z. Rather than starting at initial condition of v_z=0, we start at w≠0, and see how w approaches|decays to 0.

$$\frac{d}{dt} w = \frac{1}{r} \cdot \frac{d}{dr}\left(r \cdot \frac{d}{dr} w\right)$$

B.C. at $t = 0$ $w = 1 - r^2$ $\qquad w_{t0}(r) := v_{z0}(r) - v_{ss}(r)$
at $r = 1$ $w = 0$

Step 4. Find solution. Almost all PDE problems have no "neat" analytical solution, and we have to settle for an approximation or a numeric one. Failing to obtain a "neat" analytical expression, we express w as a linear combination of linearly independent basis vectors. Although the basis vectors can be random (as long as they are linearly independent -- even the old-fashioned power series {1, r, r^2,...} will do, but it is way too complicated), we almost always prefer orthogonal eigenvectors w_i, where each eigenvector w_i satisfies the usual eigenvalue-eigenvector relationship. Note that we have two different linear operators $\mathcal{L}_{\mathcal{T}}$ & $\mathcal{L}_{\mathcal{R}}$ for a PDE of two independent variables t and r; whereas, we have one linear operator for an ODE of one independent variable.

General eigenvalue-eigenvector equation $\mathcal{L}_{\mathcal{T}} w = \mathcal{L}_{\mathcal{R}} \cdot w = \lambda' \cdot w$

where $\mathcal{L}_{\mathcal{T}} = \frac{d}{dt}\cdot$ $\quad \mathcal{L}_{\mathcal{R}} = \frac{1}{r}\cdot\frac{d}{dr}\left(r\cdot\frac{d}{dr}\cdot\right)$ ... two linear operators

First, find eigenvalues-eigenvectors; there are infinitely many in this PDE problem.

$$\mathcal{L}_{\mathcal{T}} w_i = \mathcal{L}_{\mathcal{R}} \cdot w_i = \lambda'_i \cdot w_i$$

Then, express w as a linear combination of eigenvectors.

$$w = \sum_{i=0}^{\infty} A_i \cdot w_i = A_0 \cdot w_0 + A_1 \cdot w_1 + \ldots$$

We solve the eigenvalue-eigenvector problem for each of the two different operators.

eigenvector for the $\mathcal{L}_{\mathcal{T}}$ operator: $\mathcal{L}_{\mathcal{T}} T = \frac{d}{dt} T = \lambda' \cdot T$

eigenvector for the $\mathcal{L}_{\mathcal{R}}$ operator: $\mathcal{L}_{\mathcal{R}} R = \left[\frac{1}{r}\cdot\frac{d}{dr}\left(r\cdot\frac{d}{dr}\cdot\right)\right]\cdot R = \lambda' \cdot R$

Note that the above two are *not* equal. $\mathcal{L}_{\mathcal{T}} T \neq \mathcal{L}_{\mathcal{R}} R \qquad \lambda' \cdot T \neq \lambda' \cdot R$

Thus, neither T nor R is the solution. Remember, the general eigenvalue-eigenvector satisfies the following.

$$\mathcal{L}_{\mathcal{T}} w = \mathcal{L}_{\mathcal{R}} w = \lambda' \cdot w$$

The output from these two operators are made equal by multiplying each with the eigenvector from the other operator.

$$\begin{matrix} \left(\mathcal{L}_t \cdot T\right)\cdot R = (\lambda' \cdot T)\cdot R \\ \left(\mathcal{L}_r \cdot R\right)\cdot T = (\lambda' \cdot R)\cdot T \end{matrix} \longrightarrow \left(\mathcal{L}_t \cdot T\right)\cdot R = \lambda' \cdot T \cdot R = \left(\mathcal{L}_r \cdot R\right)\cdot T \longrightarrow \mathcal{L}_t \cdot (T\cdot R) = \mathcal{L}_r \cdot (T\cdot R) = \lambda' \cdot (T \cdot R)$$

It follows that if we let w=T·R, then w=T·R above is an eigenvector common to both operators $\mathcal{L}_{\mathcal{T}}$ & $\mathcal{L}_{\mathcal{R}}$.

$\mathcal{L}_{\mathcal{T}} w = \mathcal{L}_{\mathcal{R}} w = \lambda' \cdot w$ where $w = T \cdot R$

We repeat the same for each eigenvalue λ'_i and obtain for each eigenvalue a corresponding eigenvector.

for λ'_i $\quad w_i = T_i \cdot R_i$

The above *derives* (not assumes) that w is a product of T & R, and is exactly equivalent to the common method of PDE solution via **separation of variables**, where we *assume* w(t,r) is a product of two separate functions T(t) and R(r), where T(t) is a function of only t, and R(r) is a function of only r.

$$w(t,r) = T(t)\cdot R(r)$$

Substituting the above product into the PDE yields,

$$\left(\frac{d}{dt}T\right)\cdot R = T\cdot\left[\frac{1}{r}\cdot\frac{d}{dr}\left(r\cdot\frac{d}{dr}R\right)\right] \longrightarrow \frac{1}{T}\cdot\frac{d}{dt}T = \frac{1}{R}\cdot\left[\frac{1}{r}\cdot\frac{d}{dr}\left(r\cdot\frac{d}{dr}R\right)\right]$$

The LHS is a function of t only, and the RHS is a function of r only. For these two statements to hold, the only possibility is for both the LHS and the RHS to be a constant λ'. This give rise to a couple of **eigenvalue-eigenvector** equations -- the same thing we arrived at via the linear transformation approach.

$$\frac{1}{T}\cdot\frac{d}{dt}T=\lambda'=\frac{1}{R}\cdot\left[\frac{1}{r}\cdot\frac{d}{dr}\left(r\cdot\frac{d}{dr}R\right)\right]$$

LHS: $\frac{1}{T}\cdot\frac{d}{dt}T=\lambda'$ $\longrightarrow$ $\mathcal{L}_t\cdot T=\frac{d}{dt}T=\lambda'\cdot T$ Eq (1) ... eigenvalue-eigenvector of a first derivative operator $\mathcal{L}_q$.

RHS: $\frac{1}{R}\cdot\left[\frac{1}{r}\cdot\frac{d}{dr}\left(r\cdot\frac{d}{dr}R\right)\right]=\lambda'$ $\longrightarrow$ $\mathcal{L}_r\cdot R=\frac{1}{r}\cdot\frac{d}{dr}\left(r\cdot\frac{d}{dr}R\right)=\lambda'\cdot R$ Eq (2) ... eigenvalue-eigenvector of a second derivative operator $\mathcal{L}_{\mathcal{R}}$.

Solution of eigenvectors T in Eq (1).

$$T(t)=A\cdot\exp(\lambda'\cdot t)$$

Note that although any value of λ' is an eigenvalue to the first derivative operator $\mathcal{L}_q$ (and any value of λ' is an eigenvalue for $\mathcal{L}_q$), consideration of the boundary condition in Eq (2) will allow only certain λ' to be valid.

Solution of eigenvectors R in Eq (2).

$$\frac{1}{r}\cdot\frac{d}{dr}\left(r\cdot\frac{d}{dr}R\right)=\lambda'\cdot R=-\lambda^2\cdot R$$ where, for convenience, $\lambda'=-\lambda^2$

B.C. at $r=0$ $R=\text{bound}$

at $r=1$ $R=0$

$$r^2\cdot\frac{d^2}{dr^2}R+r\cdot\frac{d}{dr}R+\lambda^2\cdot r^2\cdot R=0$$

Let $x=\lambda\cdot r$ $\longrightarrow$ $x^2\cdot\frac{d^2}{dx^2}R+x\cdot\frac{d}{dx}R+x^2\cdot R=0$ ... Bessel's differential equation of order 0

The solutions are Bessel's function of the first kind of order 0 (J_0) and Bessel's function of the second kind of order 0 (Y_0).

$$R=A\cdot J_0(x)+B\cdot Y_0(x)=A\cdot J_0(\lambda\cdot r)+B\cdot Y_0(\lambda\cdot r)$$

Evaluate constants (including eigenvalues) from boundary conditions.

Boundary Condition #0. Bound solution at r=0 requires B=0, because $Y_0(0)=-\infty$. $R(r)=A\cdot J_0(\lambda\cdot r)$

Boundary Condition #1. At $r=1$ $w=0$ This is where we pin down the eigenvalues and eigenfunctions.

$w(t,1)=0=A\cdot\exp\left(-\lambda^2\cdot t\right)\cdot J_0(\lambda\cdot 1)$ Since exponential is never 0, $0=J_0(\lambda\cdot 1)=J_0(\lambda)$

Thus, the eigenvalues correspond to the roots of the Bessel's function J_0. Note that because the function J_0 oscillates around 0 (much like the cosine function), there are infinitely many eigenvalues (λ_i, i=0,1,2,...) and the associated eigenfunctions.

Evaluate the eigenvalues.

Define a root-finding function that allows an initial guess to be specified. $\text{rootJ0}(x):=\text{root}(\text{J0}(x),x)$

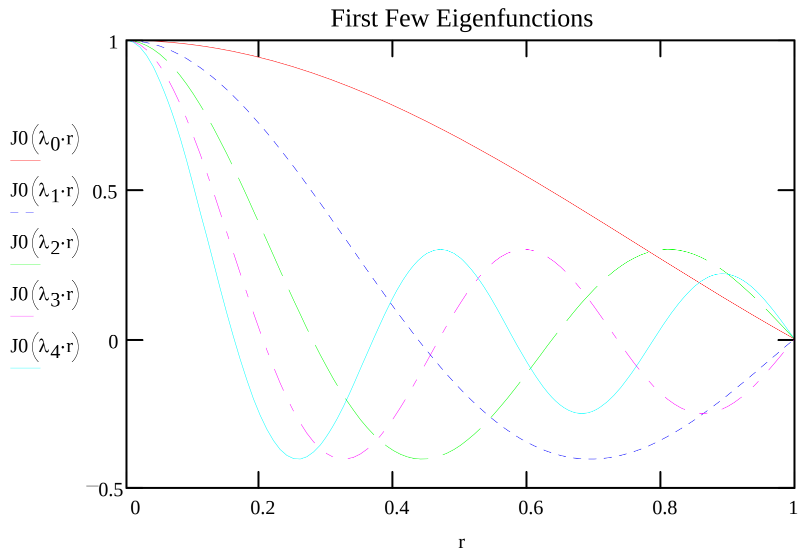
$n:=10$ $i:=0..n-1$ Start with the first root: $\lambda_0:=\text{rootJ0}(1)$

Add π to obtain an initial guess for the subsequent roots: $\lambda_{i+1}:=\text{rootJ0}\left(\lambda_i+\pi\right)$

$\lambda^T=$

	0	1	2	3	4	5	6	7	8	9	10
0	2.4048	5.5200	8.6537	11.7915	14.9309	18.0711	21.2116	24.3525	27.4935	30.6346	33.7758

$r:=0,0.01..1$



Each eigenfunction satisfies the given PDE and the boundary condition w(t,1)=0.

$$w_i(t,r)=\exp\left[-(\lambda_i)^2\cdot t\right]\cdot J_0(\lambda_i\cdot r) \qquad \text{for} \qquad i=0,1,\ldots$$

Furthermore, any scalar multiple of an eigenfunction is also an eigenfunction. Thus, any linear combination of the eigenfunctions also satisfies the given PDE along with Boundary Condition #1.

$$w(t,r)=\sum_{i=0}^{\infty} A_i\cdot w_i(t,r) = \sum_{i=0}^{\infty} A_i\cdot\exp\left[-(\lambda_i)^2\cdot t\right]\cdot J_{(i)(r)} = \sum_{i=0}^{\infty} A_i\cdot\exp\left[-(\lambda_i)^2\cdot t\right]\cdot J_0(\lambda_i\cdot r)$$

To repeat, w is a linear combination of eigenvectors w_i. Furthermore, the above equation contains the following elements:

A_i	... approximates $w_{t0}=1-r^2$, *initial condition* for the deviation variable w
$\exp\left[-(\lambda_i)^2\cdot t\right]$	... describes decay to steady-state w=0; each eigenfunction J component decays with a different time constant. In other words, each mode has its own dynamics that is independent on other modes -- another reason for working with eigenfunctions. The eigenfunctions (J) themselves, being the basis, remain unchanged; the magnitude of the component (A) decays.
$J_i=J_0(\lambda_i\cdot r)$	... eigenvectors, where eigenvalues λ_i chosen to satisfy the *boundary condition* at the pipe wall r=1

Boundary Condition #2. At $t=0$ $w=w_{t0}=1-r^2$ Based on this, we find the coefficients A_i.

$$w(0,r)=w_{t0}(r)=1-r^2=\sum_{i=0}^{\infty} A_i\cdot\exp\left[-(\lambda_i)^2\cdot 0\right]\cdot J_0(\lambda_i\cdot r) = \sum_{i=0}^{\infty} A_i\cdot J_0(\lambda_i\cdot r) = A_0\cdot J_0(\lambda_0\cdot r)+A_1\cdot J_0(\lambda_1\cdot r)+\ldots$$

Thus, in plain English, we wish to express a given function $w_{t0}=1-r^2$ with a series of basis functions (i.e., Bessel's functions J_0 in this case). We find the coefficients A_i from the projection. If the basis functions are orthogonal, each term is decoupled from other terms, and we can evaluate each term independent of other terms, meaning no matrix inverse. Projection is simply the scalar product of the function to be approximated and the Bessel basis vectors.

$$A_i = \frac{\left(w_{t0}(r), J_0\left(\lambda_i \cdot r\right)\right)}{\left(J_0\left(\lambda_i \cdot r\right), J_0\left(\lambda_i \cdot r\right)\right)}$$ ... approximate $w_{t0}(r)=w(0,r)=1-r^2$ with basis J_0 via projection

Note: for a problem that starts with an initial velocity profile of $v_z(t=0,r)=v_{z0}(r)\neq 0$

The initial deviation variable w is $w(t=0,r)=w_{t0}(r)=v_{z0}(r)-v_{ss}(r)$

We simply approximate w(0,r), instead of $w(0,r)=1-r^2$, with Bessel basis vectors.

Numerical Evaluation. Although some people consider this approach analytical, it eventually comes down to numerical evaluation, because it is difficult to visualize the solution when there are many terms that make up the solution. First, we need to define a convenient scalar product. (Note that other scalar product definition will also do, but the projection formula becomes a bit more complicated. With the following definition involving a weighting function of r, the Bessel's functions are mutually orthogonal (but not normalized).

Define scalar product: $$\mathrm{prod}(f,g) := \int_0^1 r \cdot f(r) \cdot g(r)\, dr$$

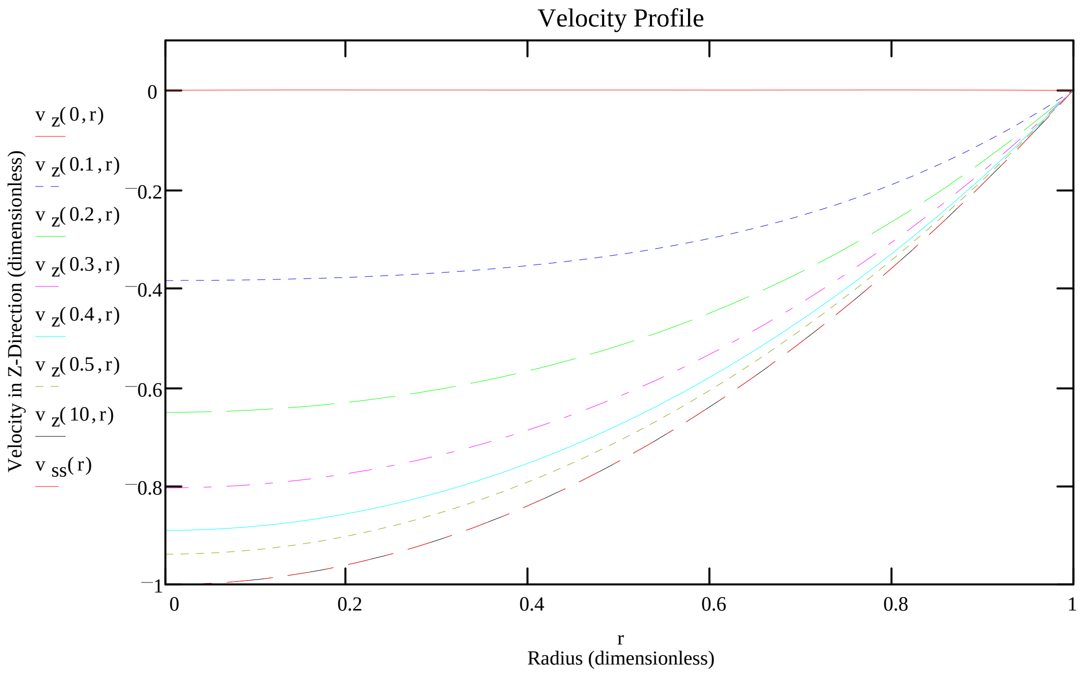
Function to approximate via projection based on the above scalar product: $w_{t0}(r)$

$i := 0 .. n$

$$A_i := \frac{\int_0^1 r \cdot w_{t0}(r) \cdot J0\left(\lambda_i \cdot r\right) dr}{\int_0^1 r \cdot J0\left(\lambda_i \cdot r\right) \cdot J0\left(\lambda_i \cdot r\right) dr}$$ ⟵ Approximate w_{t0} with Fourier Bessel series

$$w(t,r) := \sum_{i=0}^{n} A_i \cdot \exp\left[-\left(\lambda_i\right)^2 \cdot t\right] \cdot J0\left(\lambda_i \cdot r\right)$$ ⟵ Change the upper limit to see the effect of the number of terms. In fact, we do a good job with just one single term if we were to tolerate error at t=0. Try it!

Finally, the dimensionless z-direction velocity is: $v_z(t,r) := w(t,r) + v_{ss}(r)$



t=0 (initial condition)
t=0.1
t=0.2
t=0.3
t=0.4
t=0.5
t=10 (practically steady-state)
Steady-state

Side Notes. Steady-state solution. It is usually easier to shift the variables so that "0" represents the steady-state and we study how the variable(s) approaches|decays toward steady-state.

Analogy -- linearization around steady-state in PDE vs linearization around steady-state in local linearized stability analysis of ODEs. In solving a set of nonlinear autonomous first-order ODEs, we first linearize around each steady-state value x_{ss}:

Given "dx/dt=f(x)" $$\frac{d}{dt}x = f(x) = f(x_{ss}) + \left(\frac{df}{dx}\right)_{x_{ss}}\cdot(x - x_{ss}) + \frac{1}{2}\cdot(x - x_{ss})^T\cdot\left(\frac{d2f}{dx^2}\right)_{x_{ss}}\cdot(x - x_{ss}) + ...$$

After ignoring higher order terms: $$\frac{d}{dt}x = f(x) = f(x_{ss}) + \left(\frac{df}{dx}\right)_{x_{ss}}\cdot(x - x_{ss}) \qquad \frac{d}{dt}x = \frac{d}{dt}(x - x_{ss})$$

In deviation variables: $$\frac{dX}{dt} = A\cdot x \quad \text{where} \quad f(x_{ss}) = 0 \quad A = \left(\frac{df}{dx}\right)_{x_{ss}} \quad X = x - x_{ss}$$

We then find eigenvalue-eigenvector for $A\cdot x = \lambda\cdot x \quad A\cdot V = V\cdot\Lambda \quad X = V\cdot\exp(J\cdot t)\cdot V^{-1}\cdot X_0 = \exp(A\cdot t)\cdot X_0$

We finally examine the eigenvalues λ to determine stability by virtue of how the deviation variable X eventually settles from X_0 to the origin X=0, and we examine the eigenvectors to find phase trajectories (which are attracted toward the origin X=0 if stable, or repulsed away from the origin X=0 if unstable). Here in solving a PDE, we examine how the deviation variable w eventually settles from w_{t0} to the origin w=0.

In this **PDE solution via separation of variables**, we also follow the same procedure by which we find the steady-state first, then linearize around the steady-state with a deviation variable. This particular problem is a linear one (with linear operators|transformations $\mathcal{L}_t$ & $\mathcal{L}_r$). However, had the problem been a nonlinear one, we would execute the linearization step, and this linearization approach would remain the method we take as a first approximation.

Steady-state solution. At steady state, d/dt=0

$$\frac{d}{dt}v_z = 0 = \left(\mathcal{NL}_{\mathcal{R}}\cdot v_z\right)_{steady\_state} = -4 + \frac{1}{r}\cdot\frac{d}{dr}\left(r\cdot\frac{d}{dr}v_{ss}\right)$$

B.C. at $t=0$ $v_z=0$

at $r=1$ $v_z=0$

$$\frac{d}{dr}\left(r\cdot\frac{d}{dr}v_{ss}\right) = 4\cdot r \longrightarrow \int 1\, d\left(r\cdot\frac{d}{dr}v_{ss}\right) = \int 4\cdot r\, dr$$

$r\cdot\frac{d}{dr}v_{ss} = 2\cdot r^2 + C_1$ $\longrightarrow$ apply BC at r=0 (symmetry) $\frac{d}{dr}v_{ss} = 0 \longrightarrow C_1 = 0 \qquad \frac{d}{dr}v_{ss} = 2\cdot r$

$v_{ss} = \int 2\cdot r\, dr = r^2 + C_2$ $\longrightarrow$ apply BC at r=1 $v_{ss} = 0 \longrightarrow C_2 = 1 \quad v_{ss}(r) = r^2 - 1$ steady state solution

Linearization to convert the original non-linear transformation ($\mathcal{NL}_{\mathcal{T}}$ or $\mathcal{NL}_{\mathcal{R}}$) into a linear transformation ($\mathcal{L}_{\mathcal{T}}$ or $\mathcal{L}_{\mathcal{R}}$) . The term "4" makes the original transformation nonlinear.

Analogy: $\mathcal{NL}\cdot x = A\cdot x + b$ is a nonlinear transformation because:

$$\mathcal{NL}\cdot(x + y) = A\cdot(x + y) + b = (A\cdot x + b) + (A\cdot y + b) - b = \mathcal{NL}\cdot(x) + \mathcal{NL}\cdot(y) - b$$

$\mathcal{NL}\cdot(x + y) \neq \mathcal{NL}\cdot(x) + \mathcal{NL}\cdot(y)$ ... violate the definition of a linear transformation, unless b=0

Original: nonlinear transformation $\frac{d}{dt}v_z = -4 + \frac{1}{r}\cdot\frac{d}{dr}\left(r\cdot\frac{d}{dr}v_z\right)$

$\mathcal{NL}_{\mathcal{R}}\cdot(v) = -4 + \frac{1}{r}\cdot\frac{d}{dr}\left(r\cdot\frac{d}{dr}v\right)$ is a nonlinear transformation because:

$$\mathcal{NL}_{\mathcal{R}}\cdot(v+w) = -4 + \frac{1}{r}\cdot\frac{d}{dr}\left[r\cdot\frac{d}{dr}(v+w)\right] = \left[-4 + \frac{1}{r}\cdot\frac{d}{dr}\left(r\cdot\frac{d}{dr}v\right)\right] + \left[-4 + \frac{1}{r}\cdot\frac{d}{dr}\left(r\cdot\frac{d}{dr}w\right)\right] + 4 = \mathcal{NL}_{\mathcal{R}}\cdot(v) + \mathcal{NL}_{\mathcal{R}}\cdot(w) + 4$$

$\mathcal{NL}_{\mathcal{R}}\cdot(v+w) \neq \mathcal{NL}_{\mathcal{R}}\cdot(v) + \mathcal{NL}_{\mathcal{R}}\cdot(w)$ ... violate the definition of a linear transformation

Shifting the term "-4" to LHS does not circumvent the nonlinear transformation problem, because doing so merely shifts the nonlinearity to the other transformation.

$\mathcal{NL}_{\mathcal{T}}\cdot(v) = \frac{d}{dt}v_z + 4$ is a nonlinear transformation because:

$$\mathcal{NL}_{\mathcal{T}}\cdot(v+w) = \frac{d}{dt}(v+w) + 4 = \left(\frac{d}{dt}v + 4\right) + \left(\frac{d}{dt}w + 4\right) + 4 = \mathcal{NL}_{\mathcal{T}}\cdot(v) + \mathcal{NL}_{\mathcal{T}}\cdot(w) - 4$$

$\mathcal{NL}_{\mathcal{T}}\cdot(v+w) \neq \mathcal{NL}_{\mathcal{T}}\cdot(v) + \mathcal{NL}_{\mathcal{T}}\cdot(w)$ ... violate the definition of a linear transformation

Since we must have linear transformations to even apply the eigenvalue-eigenvector idea, this nonlinearization step is a must. In this PDE when we work with a deviation variable $w=v_z-v_{ss}$ that is the difference between the original variable and its steady-state value, we also in effect linearize the original nonlinear transformation $\mathcal{NL}_{\mathcal{R}}$ into a linear one $\mathcal{L}_{\mathcal{R}}\cdot$.

Given "dx/dt=f(x)" $\frac{dx}{dt} = f(x) = f(x_{ss}) + \left(\frac{df}{dx}\right)_{x_{ss}}\cdot(x - x_{ss}) + \frac{1}{2}\cdot(x - x_{ss})^T\cdot\left(\frac{d2f}{dx^2}\right)_{x_{ss}}\cdot(x - x_{ss}) + ...$

Given "$dv_z/dt=\mathcal{NL}_{\mathcal{R}}\cdot v_z$" $\frac{d}{dt}v_z = \mathcal{NL}_{\mathcal{R}}\cdot v_z = \mathcal{NL}_{\mathcal{R}}\cdot v_{ss} + \mathcal{L}_{\mathcal{R}}\cdot(v_z - v_{ss}) + ...\text{higher\_order\_terms}$

In solving for the steady-state solution, we have $\mathcal{NL}_{\mathcal{R}}\cdot v_{ss} = 0$

$\frac{d}{dt}(v_z) = \frac{d}{dt}(v_z - v_{ss})$

After ignoring higher order terms (there happen to be none in this PDE):

$$\frac{d}{dt}(v_z - v_{ss}) = 0 + \mathcal{L}_{\mathcal{R}}\cdot(v_z - v_{ss})$$

In deviation variables: $\frac{d}{dt}w = \mathcal{L}_{\mathcal{R}}\cdot w$ where $w = v_z - x_{ss}$

Analogy -- steady-state solution of PDE vs solution of Linear Algebraic Equation "A·x=b".
When solve a linear algebraic equation of the standard form A·x=b. Most students start by thinking "A·x=b" as a linear algebraic equation problem, because that is the way the linear equation is usually first introduced. However, the same problem can be viewed from many perspectives, one of them being the vector perspective. From the vector perspective, with the equation "A·x=b", we are solving a projection problem where we find the coefficients x_i to represent the given vector b as a linear combination of the individual vectors in A.

perspective of "standard" linear algebraic equation: $A\cdot x = b$

projection perspective: $b = x_0 \cdot A^{\langle 0 \rangle} + x_1 \cdot A^{\langle 1 \rangle} + \dots + x_n \cdot A^{\langle n \rangle} = \left(A^{\langle 0 \rangle} \quad A^{\langle 1 \rangle} \quad \dots \quad A^{\langle n \rangle} \right) \cdot \begin{bmatrix} x_0 \\ x_1 \\ \dots \\ x_n \end{bmatrix}$

Thus, we can regard an iterative algorithm, such as Gauss-Seidel's, as an eigenvalue-eigenvector problem with an eigenvalue of $\lambda=1$.

Gauss-Seidel: $x = A \cdot x$ eigenvalue-eigenvector perspective $\longrightarrow$ $A \cdot x = \lambda \cdot x$ with $\lambda = 1$

Analogy -- steady-state solution of PDE vs solution of nonlinear Algebraic Equation "f(x)=0" with Newton's method". In Newton's method, we linearize f(x) around an initial guess x_0:

Given "f(x)=0" $f(x) = 0 = f(x_0) + \left(\frac{df}{dx}\right)_{x_0} \cdot (x - x_0) + \frac{1}{2} \cdot \left(\frac{d2f}{dx^2}\right)_{x_0} \cdot (x - x_0)^2 + \dots$

After ignoring higher order terms: $f(x) - f(x_0) = \left(\frac{df}{dx}\right)_{x_0} \cdot (x - x_0)$

In deviation variables: $F = f_x \cdot \xi$ where $F = f(x) - f(x_0)$ $\quad f_x = \left(\frac{df}{dx}\right)_{x_0}$ $\quad \xi = (x - x_0)$

Thus, we can regard this nonlinear equation solution process as first linearizing around the current|initial guess x_0, and we represent the eventual value as x, and we work in terms of the deviation variables. And as in the standard linear algebraic equation above, with the standard nonlinear equation, we regard the solution of the resulting linear algebraic equation either as a vector projection problem or as an eigenvalue-eigenvector problem.

Analogy: vector projection -- express F as a linear combination of the basis vectors $f_x^{\langle i \rangle}$

$$F = f_x \cdot \xi = \left(f_x^{\langle 0 \rangle} \quad f_x^{\langle 1 \rangle} \quad \dots \quad f_x^{\langle n \rangle} \right) \cdot \begin{bmatrix} \xi_0 \\ \xi_1 \\ \dots \\ \xi_n \end{bmatrix} = \xi_0 \cdot f_x^{\langle 0 \rangle} + \xi_1 \cdot f_x^{\langle 1 \rangle} + \dots + \xi_n \cdot f_x^{\langle n \rangle}$$

Basically with Newton's iterative procedure or any other iterative procedure for solving nonlinear algebraic equations, we are in effect solving an eigenvalue-eigenvector problem with an eigenvalue of $\lambda=1$.

Newton's iteration scheme with linearized approximation: $x_{i+1} = x_i - \left[\left(\frac{df}{dx}\right)_{(x_i)} \right]^{-1} \cdot f(x_i) = g(x_i)$

perspective of iteration algorithm: $x = g(x)$

eigenvalue-eigenvector perspective for the linear operator $\mathcal{G}$ $\longrightarrow$ $\mathcal{G} \cdot x = g(x) = \lambda \cdot x$ with $\lambda = 1$

With an iterative procedure such as Newton's method, we find a set of coefficients ξ_i that describe the given initial deviation vector|function F. The dynamics of an iterative algorithm is such that the coefficient ξ_i eventually settle to a steady-state solution where the deviation independent variable ξ and the dependent variable F eventually become 0. Compare the above to our PDE problem here, where we find the coefficients A_i to describe the given initial deviation vector|function w_{t0}. The dynamics eventually settle to a steady-state solution where the deviation variable w become 0. In summary, the next step in PDE solution is representation of the steady-state solution in the form of a deviation variable as a linear combination of basis vectors|functions.

Analogy with linear algebraic equations "Ax=b"

$$b = A\cdot x = \left(A^{\langle 0\rangle} \quad A^{\langle 1\rangle} \quad \ldots \quad A^{\langle n\rangle}\right)\cdot\begin{bmatrix} x_0 \\ x_1 \\ \ldots \\ x_n \end{bmatrix} = x_0\cdot A^{\langle 0\rangle} + x_1\cdot A^{\langle 1\rangle} + \ldots + x_n\cdot A^{\langle n\rangle}$$

Analogy with nonlinear algebraic equations "f(x)=0"

$$F = f_x\cdot\xi = \left(f_x^{\langle 0\rangle} \quad f_x^{\langle 1\rangle} \quad \ldots \quad f_x^{\langle n\rangle}\right)\cdot\begin{bmatrix} \xi_0 \\ \xi_1 \\ \ldots \\ \xi_n \end{bmatrix} = \xi_0\cdot f_x^{\langle 0\rangle} + \xi_1\cdot f_x^{\langle 1\rangle} + \ldots + \xi_n\cdot f_x^{\langle n\rangle}$$

Analogy: vector projection -- in this PDE we represent the steady-state solution w_{t0} as a linear combination of the basis vectors|functions $J_0(\lambda_i\cdot r)$

$$w_{t0} = J\cdot A = \left(J_0(\lambda_0\cdot r) \quad J_0(\lambda_1\cdot r) \quad \ldots \quad J_0(\lambda_n\cdot r)\right)\cdot\begin{bmatrix} A_0 \\ A_1 \\ \ldots \\ A_n \end{bmatrix} = A_0\cdot J_0(\lambda_0\cdot r) + A_1\cdot J_0(\lambda_1\cdot r) + \ldots + A_n\cdot J_0(\lambda_n\cdot r)$$

$$w_{t0} = A\cdot J = \left(A_0 \quad A_1 \quad \ldots \quad A_n\right)\cdot\begin{bmatrix} J_0(\lambda_0\cdot r) \\ J_0(\lambda_1\cdot r) \\ \ldots \\ J_0(\lambda_n\cdot r) \end{bmatrix} = A_0\cdot J_0(\lambda_0\cdot r) + A_1\cdot J_0(\lambda_1\cdot r) + \ldots + A_n\cdot J_0(\lambda_n\cdot r)$$

Side Note. Analogy of dynamic solution of PDE vs solution of ODE. We compare z (ODE) vs w (PDE), initial states z_0 (ODE) vs w_{t0} (PDE), and exponential approach from z_0 to z=0 (ODE) vs exponential approach from w_{t0} to w=0 (PDE). The exponential approach to 0 in either the ODE case and the present PDE case is based on the eigenvalues as the speed of approach (i.e., the time constants of the dynamic process).

$$w = A_0\cdot\exp(\lambda'_0\cdot t)\cdot J_0(\lambda_0\cdot r) + A_1\cdot\exp(\lambda'_1\cdot t)\cdot J_0(\lambda_1\cdot r) + \ldots + A_n\cdot\exp(\lambda'_n\cdot t)\cdot J_0(\lambda_n\cdot r) \qquad \lambda' = -\lambda^2$$

$$w = \exp(\lambda'_0\cdot t)\cdot(A_0\cdot J_0) + \exp(\lambda'_1\cdot t)\cdot(A_1\cdot J_1) + \ldots + \exp(\lambda'_n\cdot t)\cdot(A_n\cdot J_n)$$

PDE ⟵ compare to first-order ODEs ⟶ Non-orthogonal basis Orthogonal basis

$$\frac{d}{dt}w = \mathcal{L}_{\mathcal{R}}w \qquad \frac{dx}{dt} = A\cdot x \qquad \frac{d}{dt}z = \Lambda\cdot z$$

where $A\cdot V = V\cdot\Lambda$

$x = V\cdot z \qquad z = V^{-1}\cdot x$

$$\frac{d}{dt}w_i = \mathcal{L}_{\mathcal{T}}w_i = \lambda'_i\cdot w_i \qquad \frac{d}{dt}z_i = A\cdot z_i = \lambda_i\cdot z_i$$

$$w_i = \exp(\lambda'_i\cdot t)\cdot w_{t0_i} \qquad z_i = \exp(\lambda_i\cdot t)\cdot z_{0_i}$$

$$w = \exp(\Lambda\cdot t)\cdot w_{t0} \qquad x = \exp(A\cdot t)\cdot x_0 \qquad z = \exp(\Lambda\cdot t)\cdot z_0 \qquad \exp(A\cdot t) = V\cdot\exp(\Lambda\cdot t)\cdot V^{-1}$$

In this PDE problem the overall vector w is decomposed into different modes|eigenvectors J_0 of different "frequencies"; whereas, in the ODE problem, z is composed of different individual states in the eigenvector coordinates; each mode has a characteristic time constant λ'_i that is independent of the other modes (i.e., the different modes are non-interacting). In other words, we break up a given function w_{t0} into different parts then "sort" different parts into different bins according to their time constants, with the high frequency components (say, $J_0(\lambda_{10}\cdot t)$) decaying faster and the low frequency components (say, $J_0(\lambda_0\cdot t)$) persisting longer. Had we worked with non-orthogonal basis, the exponential decay term exp(A·t) would have been unnecessarily complicated, and each non-orthogonal component would possess a mixture of different time constants.

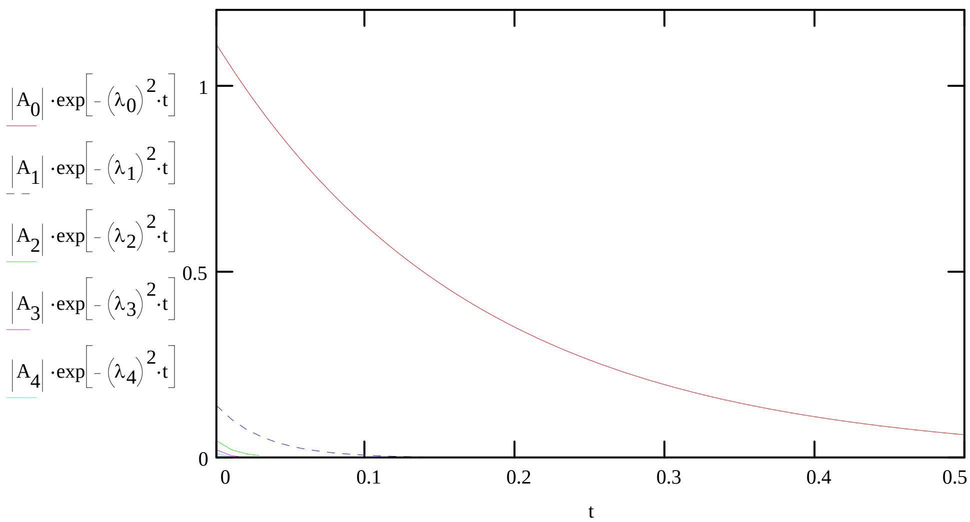
$$\exp(\Lambda\cdot t) = \begin{bmatrix} \exp(\lambda'_0\cdot t) & 0 & \ldots & 0 \\ 0 & \exp(\lambda'_1\cdot t) & \ldots & 0 \\ \ldots & \ldots & \ldots & \ldots \\ 0 & 0 & \ldots & \exp(\lambda'_n\cdot t) \end{bmatrix}$$

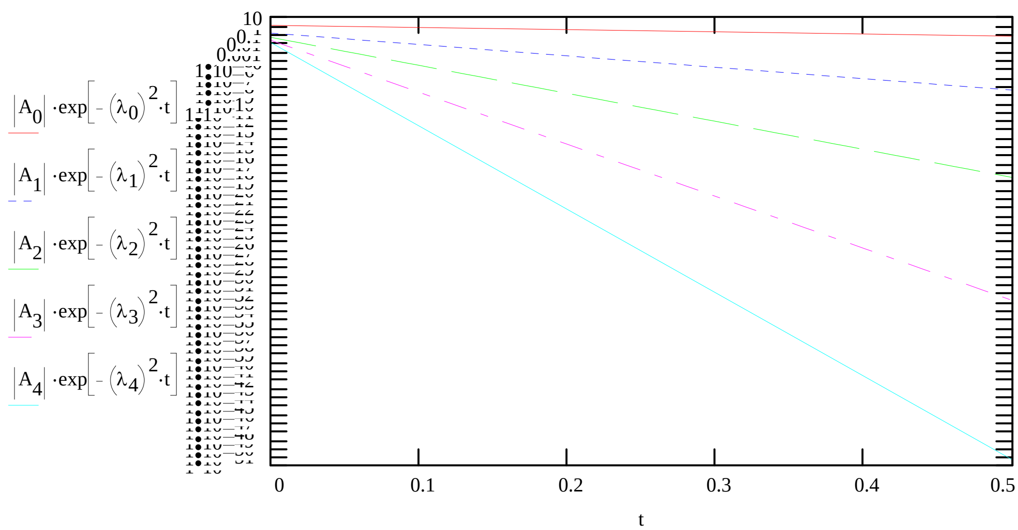
ODE $$z_0 = V^T\cdot x_0 = X_{00}\cdot\left(V^T\right)^{\langle 0\rangle} + X_{10}\cdot\left(V^T\right)^{\langle 1\rangle} + \ldots + X_{n0}\cdot\left(V^T\right)^{\langle n\rangle} = \left[\left(V^T\right)^{\langle 0\rangle} \ \left(V^T\right)^{\langle 1\rangle} \ \ldots \ \left(V^T\right)^{\langle n\rangle}\right]\cdot\begin{bmatrix} X_{00} \\ X_{10} \\ \ldots \\ X_{n0}\end{bmatrix}$$

PDE $$w_{t0} = J\cdot A = A_0\cdot J_0 + A_1\cdot J_1 + \ldots + A_n\cdot J_n = \begin{pmatrix} J_0 & J_1 & \ldots & J_n\end{pmatrix}\cdot\begin{bmatrix} A_0 \\ A_1 \\ \ldots \\ A_n\end{bmatrix}$$

	PDE $\longleftarrow$ compare $\longrightarrow$	ODE /w eigenvector basis
deviation variable	w	z
deviation variable at t=0	w_{t0}	z_0
coefficients to describe w_0 or z_0	$A_0, A_1, \ldots, A_n$	$X_{00}, X_{10}, \ldots, X_{n0}$
basis vectors	$J_0, J_1, \ldots, J_n$	$\left(V^T\right)^{\langle 0\rangle}, \left(V^T\right)^{\langle 1\rangle}, \ldots, \left(V^T\right)^{\langle n\rangle}$
exponential decay	$\exp(\Lambda \cdot t)$	$\exp(\Lambda \cdot t)$

$t := 0, 0.01 .. 0.5$





Except for A_0, all other coefficients very quickly decay to 0. That is why we need only a few terms.

Side Note. Express $w_{t0}(r)$ with non-orthogonal power series basis vectors -- just to double-check for fun. This is truly silly because we already have orthogonal basis. Working with ugly non-orthogonal basis when we already have beautiful orthogonal basis is like going for a hamburger when there is already a juicy steak, or like cheating with ugly girls when we already have a beautiful one -- just plain silly unless you enjoy deliberately making things complex.

Non-example of change-of-basis (Non-example of similarity transform) $\quad power = a \cdot bessel$

$$\begin{bmatrix} 1 \\ r \\ \cdots \\ r^n \end{bmatrix} = \begin{bmatrix} a_{00} & a_{01} & \cdots & a_{0n} \\ a_{10} & a_{11} & \cdots & a_{1n} \\ \cdots & \cdots & \cdots & \cdots \\ a_{n0} & a_{n1} & \cdots & a_{nn} \end{bmatrix} \cdot \begin{bmatrix} J0(\lambda_0 \cdot r) \\ J0(\lambda_1 \cdot r) \\ \cdots \\ J0(\lambda_n \cdot r) \end{bmatrix}$$

Evaluation of change-of-basis matrix a -- coefficients to change from Bessel series J to power series

$$\begin{bmatrix} (r^0, J0(\lambda_0 \cdot r)) & (r^1, J0(\lambda_0 \cdot r)) & \cdots & (r^n, J0(\lambda_0 \cdot r)) \\ (r^0, J0(\lambda_1 \cdot r)) & (r^1, J0(\lambda_1 \cdot r)) & \cdots & (r^n, J0(\lambda_1 \cdot r)) \\ \cdots & \cdots & \cdots & \cdots \\ (r^0, J0(\lambda_n \cdot r)) & (r^1, J0(\lambda_n \cdot r)) & \cdots & (r^n, J0(\lambda_n \cdot r)) \end{bmatrix} = \begin{bmatrix} (J0(\lambda_0 \cdot r), J0(\lambda_0 \cdot r)) & (J0(\lambda_1 \cdot r), J0(\lambda_0 \cdot r)) & \cdots & (J0(\lambda_n \cdot r), J0(\lambda_0 \cdot r)) \\ (J0(\lambda_0 \cdot r), J0(\lambda_1 \cdot r)) & (J0(\lambda_1 \cdot r), J0(\lambda_1 \cdot r)) & \cdots & (J0(\lambda_n \cdot r), J0(\lambda_1 \cdot r)) \\ \cdots & \cdots & \cdots & \cdots \\ (J0(\lambda_0 \cdot r), J0(\lambda_n \cdot r)) & (J0(\lambda_n \cdot r), J0(\lambda_1 \cdot r)) & \cdots & (J0(\lambda_n \cdot r), J0(\lambda_n \cdot r)) \end{bmatrix} \cdot \begin{bmatrix} a_{00} \\ a_{01} \\ \cdots \\ a_{0n} \end{bmatrix}$$

$rJ0 = J0J0^T \cdot a^T \qquad rJ0^T = a \cdot J0J0 \qquad a = rJ0^T \cdot J0J0^{-1}$

scalar products of r and basis J $\qquad$ matrix of scalar products of basis J $\qquad i := 0..n \qquad j := 0..n$

$$rJ0_{i,j} := \int_0^1 r \cdot r^j \cdot J0(\lambda_i \cdot r)\, dr \qquad J0J0_{i,j} := \int_0^1 r \cdot J0(\lambda_i \cdot r) \cdot J0(\lambda_j \cdot r)\, dr$$

actually only the diagonal elements are needed, because of orthogonality

$$a_{i,j} := \frac{rJ0_{j,i}}{J0J0_{j,j}}$$

Non-example of change-of-basis (Non-example of similarity transform) $\quad bessel = b \cdot power$

$$\begin{bmatrix} J0(\lambda_0 \cdot r) \\ J0(\lambda_1 \cdot r) \\ \cdots \\ J0(\lambda_n \cdot r) \end{bmatrix} = \begin{bmatrix} b_{00} & b_{01} & \cdots & b_{0n} \\ b_{10} & b_{11} & \cdots & b_{1n} \\ \cdots & \cdots & \cdots & \cdots \\ b_{n0} & b_{n1} & \cdots & b_{nn} \end{bmatrix} \cdot \begin{bmatrix} 1 \\ r \\ \cdots \\ r^n \end{bmatrix}$$

Evaluation of change-of-basis matrix b -- coefficients to change from power series J to Bessel series

$$\begin{bmatrix} (J0(\lambda_0 \cdot r), r^0) & (J0(\lambda_1 \cdot r), r^0) & \cdots & (J0(\lambda_n \cdot r), r^0) \\ (J0(\lambda_0 \cdot r), r^1) & (J0(\lambda_1 \cdot r), r^1) & \cdots & (J0(\lambda_n \cdot r), r^1) \\ \cdots & \cdots & \cdots & \cdots \\ (J0(\lambda_0 \cdot r), r^n) & (J0(\lambda_1 \cdot r), r^n) & \cdots & (J0(\lambda_n \cdot r), r^n) \end{bmatrix} = \begin{bmatrix} (r^0, r^0) & (r^1, r^0) & \cdots & (r^n, r^0) \\ (r^0, r^1) & (r^1, r^1) & \cdots & (r^0, r^1) \\ \cdots & \cdots & \cdots & \cdots \\ (r^0, r^n) & (r^1, r^n) & \cdots & (r^0, r^n) \end{bmatrix} \cdot \begin{bmatrix} b_{00} & b_{10} & \cdots & b_{n0} \\ b_{01} & b_{11} & \cdots & b_{n1} \\ \cdots & \cdots & \cdots & \cdots \\ b_{0n} & b_{1n} & \cdots & b_{nn} \end{bmatrix}$$

$J0r = RR^T \cdot b^T$ $\quad J0r^T = b \cdot RR$ $\quad b = J0r^T \cdot RR^{-1}$

scalar products of J and basis r^i matrix of scalar products of basis $\{1,r,r^2,..,r^n\}$

$$J0r_{i,j} := \int_0^1 r \cdot J0\left(\lambda_j \cdot r\right) \cdot r^i \, dr \qquad RR_{i,j} := \int_0^1 r \cdot r^i \cdot r^j \, dr$$

The diagonal elements are needed, because the power series is non-orthogonal.

$b := J0r^T \cdot RR^{-1}$

Why "non-example"? Note that, because one set of basis does not exactly dependent on the second set of basis (in fact, they are both independent of the other), we are only approximating one set with the second set. As a result, one is *not* the inverse of the other. This is not exactly a change-of-basis. This is not exactly a similarity transform.

$a \neq b^{-1} \qquad b \neq a^{-1}$

Check: the following two are similar only for the first few columns (thus, "a" is ok, but b^{-1} fails.)

a =

	0	1	2	3	4	5
0	1.602	-1.065	0.851	-0.73	0.649	-0.59
1	0.817	-1.134	0.798	-0.747	0.632	-0.597
2	0.494	-0.925	0.806	-0.709	0.637	-0.582
3	0.33	-0.73	0.755	-0.681	0.623	-0.573
4	0.235	-0.579	0.679	-0.648	0.603	-0.561
5	0.176	-0.466	0.599	-0.607	0.579	-0.546
6	0.137	-0.381	0.525	-0.562	0.551	-0.528
7	0.109	-0.316	0.459	-0.516	0.521	-0.508
8	0.089	-0.265	0.403	-0.471	0.49	-0.486
9	0.074	-0.226	0.355	-0.429	0.459	-0.464
10	0.062	-0.194	0.313	-0.391	0.429	-0.441

b^{-1} =

	0	1	2	3	4	5
0	1.602	-1.072	0.972	1.967	4.89	-8.657
1	0.817	-1.14	0.919	1.952	4.876	-8.667
2	0.494	-0.932	0.926	1.976	4.858	-8.617
3	0.33	-0.737	0.875	1.987	4.819	-8.568
4	0.235	-0.586	0.798	1.992	4.755	-8.478
5	0.176	-0.473	0.717	2.014	4.698	-8.422
6	0.137	-0.387	0.641	2.028	4.621	-8.318
7	0.109	-0.322	0.573	2.032	4.524	-8.189
8	0.089	-0.272	0.515	2.039	4.433	-8.07
9	0.074	-0.232	0.465	2.048	4.348	-7.968
10	0.062	-0.2	0.422	2.042	4.246	-7.826

Check: the following two are not similar (thus, "b" is ok, but a^{-1} fails.)

b =

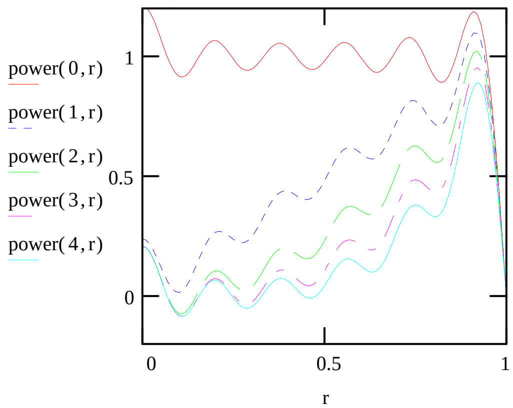
	0	1	2	3	4
0	1	0.008	-1.515	0.25	0.136
1	0.987	0.34	-10.268	8.64	3.111
2	0.891	2.818	-41.459	77.488	-14.684
3	1.077	0.256	-61.297	195.921	-135.73
4	2.482	-31.088	103.399	-71.331	-107.452
5	2.233	-35.755	174.205	-323.95	184.986
6	0.194	1.406	-25.419	71.674	-11.239
7	0.963	-15.521	73.167	-118.766	16.757
8	0.54	-9.28	53.93	-137.635	134.925
9	0.576	-9.209	43.349	-71.504	13.318
10	0.35	-6.261	37.146	-94.814	92.211

a^{-1} =

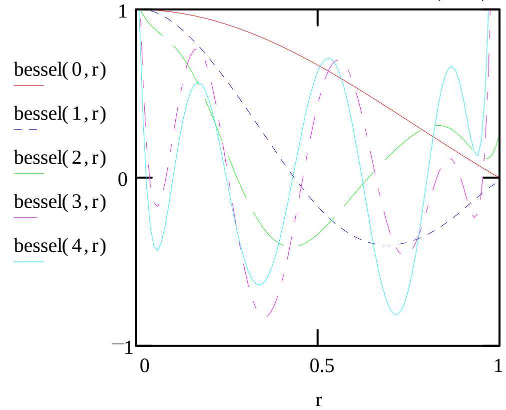
	0	1	2	3
0	1	0	-1.449	0.01
1	0.996	0.48	-14.909	40.888
2	1.062	-5.863	70.084	-508.345
3	1.905	-91.21	1345.643	-7830.291
4	1.741	-95.283	1405.387	-8214.344
5	-2.067	204.089	-3044.854	17039.069
6	-0.549	61.413	-949.002	5369.601
7	-15.489	1974.324	-30616.947	174597.879
8	-23.519	3049.668	-47489.164	271717.759
9	32.394	-3005.933	44728.773	-251209.902
10	39.274	-4007.637	60566.16	-342551.753

$$\mathrm{power}(i,r) := \sum_{j=0}^{n} a_{i,j}\cdot J0(\lambda_j\cdot r) \qquad ainv := a^{-1} \qquad \mathrm{cond2}(a) = 1.774\cdot 10^{7} \qquad \mathrm{bessel}(i,r) := \sum_{j=0}^{n} ainv_{i,j}\cdot r^{j}$$

Represent Power with Bessel (OK)

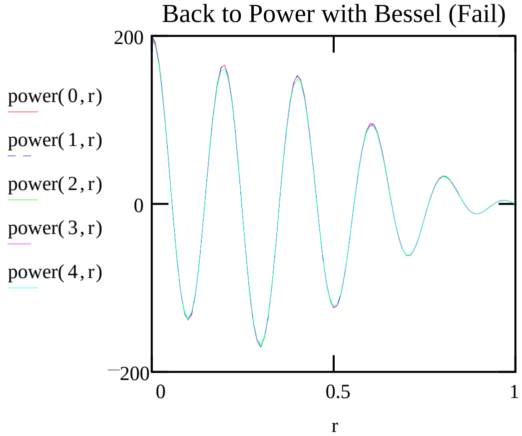


Back to Bessel with Power (Fail)

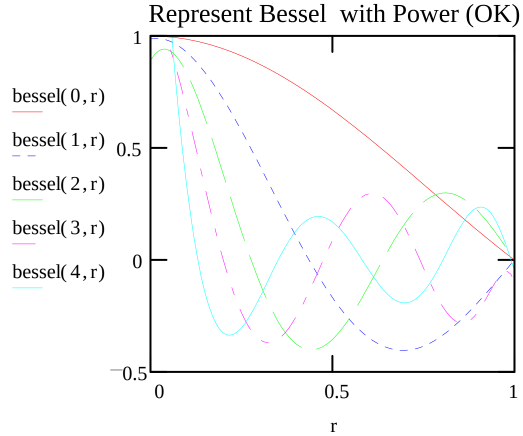


$$binv := b^{-1} \qquad \mathrm{cond2}(b) = 9.489\cdot 10^{5} \qquad \mathrm{power}(i,r) := \sum_{j=0}^{n} binv_{i,j}\cdot J0(\lambda_j\cdot r) \qquad \mathrm{bessel}(i,r) := \sum_{j=0}^{n} b_{i,j}\cdot r^{j}$$

Back to Power with Bessel (Fail)



Represent Bessel with Power (OK)



Once again, the above demonstrates that, in representing one basis with another basis, the formula involving "a" or "b" are good, but that involving a^{-1} works borderline only for the first few Bessel function and that involving b^{-1} totally fails.

Representation of w_{t0} with power series -- coefficients to change from power series J to Bessel series. (The above was representing Bessel series, instead of a given function w_{t0}, with power series.)

$$\begin{bmatrix} \left(w_{t0}(r), r^{0}\right) \\ \left(w_{t0}(r), r^{1}\right) \\ \cdots \\ \left(w_{t0}(r), r^{n}\right) \end{bmatrix} = \begin{bmatrix} \left(r^{0}, r^{0}\right) & \left(r^{1}, r^{0}\right) & \cdots & \left(r^{n}, r^{0}\right) \\ \left(r^{0}, r^{1}\right) & \left(r^{1}, r^{1}\right) & \cdots & \left(r^{0}, r^{1}\right) \\ \cdots & \cdots & \cdots & \cdots \\ \left(r^{0}, r^{n}\right) & \left(r^{1}, r^{n}\right) & \cdots & \left(r^{0}, r^{n}\right) \end{bmatrix} \cdot \begin{bmatrix} A'_{0} \\ A'_{1} \\ \cdots \\ A'_{n} \end{bmatrix} \qquad w_{t0.approx}(r) := \sum_{i=0}^{n} A_i\cdot J0(\lambda_i\cdot r)$$

$wr = RR^{T} \cdot A'$ $\quad A' = RR^{-1} \cdot wr$

scalar products of w_{t0} and basis r^{i}

$$wr_{i} := \int_{0}^{1} r \cdot w_{t0}(r) \cdot r^{i}\, dr \qquad A' := RR^{-1} \cdot wr \qquad w'_{t0.approx}(r) := \sum_{i=0}^{n} A'_{i} \cdot r^{i}$$

The above calculates the coefficients to represent w_{t0} with power series Another way of calculating A' is based on the change-of-basis matrix a that relates A to A' that approximates a given vector|function w_{t0}

Representation of w_{t0} with Bessel basis

$$w_{t0.apprix} = \begin{pmatrix} J_{0} & J_{1} & \dots & J_{n} \end{pmatrix} \cdot \begin{bmatrix} A_{0} \\ A_{1} \\ \dots \\ A_{n} \end{bmatrix}$$

Change-of-basis results in a different set of coefficients A' to describe the same vector|function w_{t0}

$$w'_{t0.apprix} = \begin{pmatrix} 1 & r & \dots & r^{n} \end{pmatrix} \cdot \begin{bmatrix} A'_{0} \\ A'_{1} \\ \dots \\ A'_{n} \end{bmatrix} = \begin{pmatrix} J_{0} & J_{1} & \dots & J_{n} \end{pmatrix} \cdot \begin{bmatrix} a_{00} & a_{10} & \dots & a_{n0} \\ a_{01} & a_{11} & \dots & a_{n1} \\ \dots & \dots & \dots & \dots \\ a_{0n} & a_{1n} & \dots & a_{nn} \end{bmatrix} \cdot \begin{bmatrix} A'_{0} \\ A'_{1} \\ \dots \\ A'_{n} \end{bmatrix}$$

Thus, the two sets of coefficients A & A' are related by the change-of-basis matrix a

$$\begin{bmatrix} A_{0} \\ A_{1} \\ \dots \\ A_{n} \end{bmatrix} = \begin{bmatrix} a_{00} & a_{10} & \dots & a_{n0} \\ a_{01} & a_{11} & \dots & a_{n1} \\ \dots & \dots & \dots & \dots \\ a_{0n} & a_{1n} & \dots & a_{nn} \end{bmatrix} \cdot \begin{bmatrix} A'_{0} \\ A'_{1} \\ \dots \\ A'_{n} \end{bmatrix}$$

$A = a^{T} \cdot A'$ ... A & A' in a column format

$A' = \left(a^{T}\right)^{-1} \cdot A$

Each method yields a different set of coefficients A to represent w_{t0} with Bessel series.

vector projection change-of-basis #1 change-of-basis #2

$A =$

	0
0	1.108
1	-0.14
2	0.045
3	-0.021
4	0.012
5	-0.007
6	0.005
7	-0.003
8	0.003
9	-0.003

$a^{T} \cdot A' =$

	0
0	1.108
1	-0.14
2	0.045
3	-0.021
4	0.012
5	-0.007
6	0.005
7	-0.003
8	0.003
9	-0.002

$\left(b^{T}\right)^{-1} \cdot A' =$

	0
0	1.108
1	-0.14
2	0.046
3	-0.008
4	0.032
5	-0.04
6	-0.019
7	-0.592
8	-0.994
9	1.013

power to Bessel ... good

$$w_{t0.approx.cob1}(r) := \sum_{i=0}^{n} \left(a^{T} \cdot A'\right)_{i} \cdot J0\left(\lambda_{i} \cdot r\right)$$

inverse(Bessel to power) ... bad

$$w_{t0.approx.cob2}(r) := \sum_{i=0}^{n} \left[\left(b^{T}\right)^{-1} \cdot A'\right]_{i} \cdot J0\left(\lambda_{i} \cdot r\right)$$

Each method yields a different set of coefficients A' to represent w_{t0} with power series.

vector projection   change-of-basis #1   change-of-basis #2

$A' =$

	0
0	1
1	0
2	-1
3	0
4	0
5	0
6	0
7	0
8	0
9	0
10	0

$b^T \cdot A =$

	0
0	0.998
1	0.035
2	-1.208
3	0.465
4	-0.358
5	0.314
6	-1.446
7	2.051
8	-0.357
9	-0.955
10	0.461

$\left(a^T\right)^{-1} \cdot A =$

	0
0	0.965
1	3.197
2	-48.544
3	266.954
4	-663.179
5	596.577
6	312.48
7	-687.867
8	-271.682
9	841.248
10	-350.282

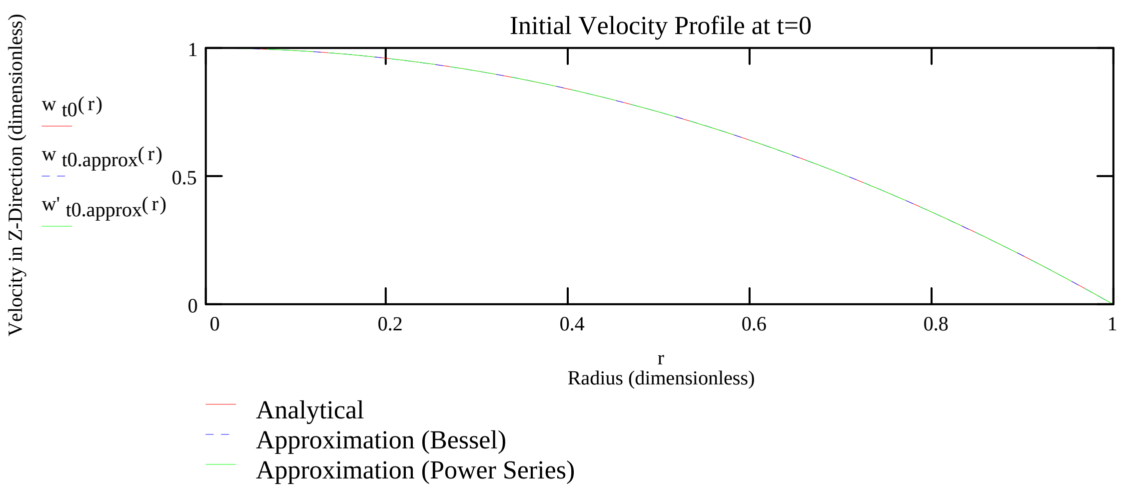
Bessel to power ... good

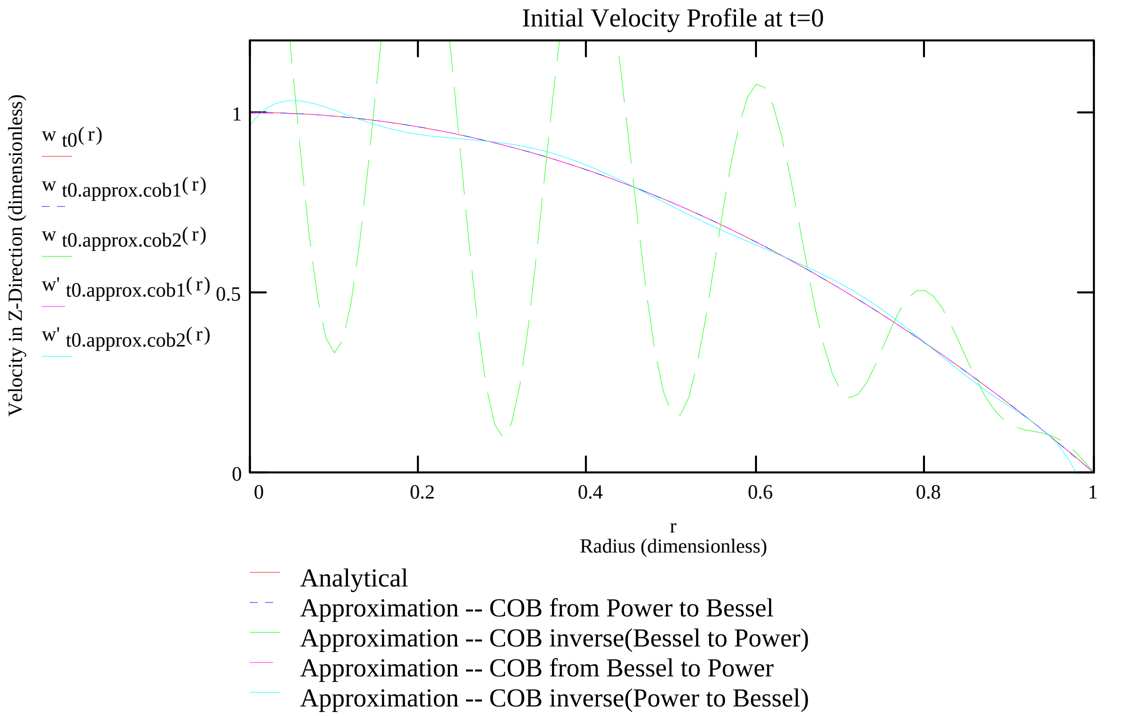
$$w'_{t0.approx.cob1}(r) := \sum_{i=0}^{n} \left(b^T \cdot A\right)_i \cdot r^i$$

inverse(power to Bessel) ... borderline ok

$$w'_{t0.approx.cob2}(r) := \sum_{i=0}^{n} \left[\left(a^T\right)^{-1} \cdot A\right]_i \cdot r^i$$

The first of the above (projection) does a descent job in approximating w_{t0}.





Once again, the above demonstrates that, similar to representing one basis with another basis, in representing a given function, the formula involving "a" or "b" are good, but that involving a^{-1} works borderline, and that involving b^{-1} fails, following the trend only roughly.

Summary: action of linear transformation $\mathcal{L}$ on basis vectors J or power

J & power in row format

$$\text{power} = J \cdot a^T$$

$$\mathcal{L} \cdot J = J \cdot \Lambda$$

$$\mathcal{L} \cdot \text{power} = J \cdot \Lambda \cdot a^T = \left[\text{power} \cdot \left(a^T\right)^{-1}\right] \cdot \Lambda \cdot a^T = \text{power} \cdot \left[\left(a^T\right)^{-1} \cdot \Lambda \cdot a^T\right]$$

J & power in column format

$$\text{power} = a \cdot J$$

$$\mathcal{L} \cdot J = \Lambda \cdot J$$

$$\mathcal{L} \cdot \text{power} = a \cdot \Lambda \cdot J = a \cdot \Lambda \cdot \left(a^{-1} \cdot \text{power}\right) = \left(a \cdot \Lambda \cdot a^{-1}\right) \cdot \text{power}$$

Matrices in "non-similarity transform" (vectors in a column format)

compare definition of a matrix L to describe the action of $\mathcal{L}$. Compare it to the line above

$\mathcal{L} \cdot \text{power}' = L \cdot \text{power}'$ ⟵ compare ⟶ $\mathcal{L} \cdot \text{power} = \left(a \cdot \Lambda \cdot a^{-1}\right) \cdot \text{power}$ ⟶ $\Lambda_{i,i} := \lambda_i$ $\quad L := a \cdot \Lambda \cdot a^{-1}$

We see that matrix "L" in this problem corresponds to the matrix "A" in the "standard" notation. Likewise, matrix "a" in this problem corresponds to the transform matrix "V" in the "standard" notation.

$L \cdot a = a \cdot \Lambda$ ... notation in this problem

$A \cdot V = V \cdot \Lambda$ ... "standard" notation in similarity transform

Eigenvalue-eigenvector for the linear transform $\mathcal{L}_{\mathcal{R}}$: $\lambda_{\text{power}} := \text{eigenvals}(L)$ $\quad V := \text{eigenvecs}(L)$

Note that Mathcad returns λ_J' that is arranged differently from λ. Likewise, V is arranged differently from a.

Rearrange in ascending order $\lambda V := \text{augment}\left(\lambda_{power}, V^{T}\right)$ $\lambda V := \text{csort}(\lambda V, 0)$

$\lambda_{power} := \text{submatrix}(\lambda V, 0, \text{rows}(\lambda V) - 1, 0, 0)$ $\Lambda_{power} := \text{diag}\left(\lambda_{power}\right)$

$V := \text{submatrix}(\lambda V, 0, \text{rows}(\lambda V) - 1, 1, \text{cols}(\lambda V) - 1)^{T}$

check: $\lambda^{T} =$

	0	1	2	3	4	5	6	7	8	9	10
0	2.405	5.52	8.654	11.792	14.931	18.071	21.212	24.352	27.493	30.635	33.776

$\lambda_{power}^{T} =$

	0	1	2	3	4	5	6	7	8	9	10
0	2.405	5.52	8.654	11.791	14.931	18.071	21.212	24.354	27.491	30.632	33.779

Mathcad returns normalized eigenvectors V. To better compare matrix "a" with matrix "V", normalize "a".

$$a_{norm}^{\langle i \rangle} := \frac{a^{\langle i \rangle}}{\sqrt{\left(a^{T} \cdot a\right)_{i,i}}}$$

check: Note that a_{norm}=V is not orthogonal, and some eigenvectors differ in sign.

$V =$

	0	1	2	3	4	5
0	0.83	0.488	0.412	0.365	0.346	0.331
1	0.424	0.519	0.386	0.374	0.337	0.336
2	0.256	0.424	0.39	0.355	0.339	0.327
3	0.171	0.334	0.365	0.341	0.332	0.322
4	0.122	0.265	0.328	0.325	0.321	0.315
5	0.091	0.213	0.29	0.304	0.308	0.307
6	0.071	0.174	0.254	0.281	0.294	0.297
7	0.057	0.145	0.222	0.258	0.278	0.285
8	0.046	0.122	0.195	0.236	0.261	0.273
9	0.038	0.103	0.171	0.215	0.245	0.261
10	0.032	0.089	0.152	0.196	0.228	0.248

$a_{norm} =$

	0	1	2	3	4	5
0	0.83	-0.488	0.412	-0.365	0.346	-0.331
1	0.424	-0.519	0.386	-0.374	0.337	-0.336
2	0.256	-0.424	0.39	-0.355	0.339	-0.327
3	0.171	-0.334	0.365	-0.341	0.332	-0.322
4	0.122	-0.265	0.328	-0.325	0.321	-0.315
5	0.091	-0.213	0.29	-0.304	0.308	-0.307
6	0.071	-0.174	0.254	-0.281	0.294	-0.297
7	0.057	-0.145	0.222	-0.258	0.278	-0.285
8	0.046	-0.122	0.195	-0.236	0.261	-0.273
9	0.038	-0.103	0.171	-0.215	0.245	-0.261
10	0.032	-0.089	0.152	-0.196	0.228	-0.248

Dynamics with non-orthogonal basis vs orthogonal basis

Analogy with 1st-order ODEs $\frac{d}{dt} z = \Lambda \cdot z$ solution $\longrightarrow$ $z = \exp(\Lambda \cdot t) \cdot z_{0}$

For PDE in this problem $\frac{d}{dt} w = \mathcal{L}_{\mathcal{R}} w$ solution $\longrightarrow$ $w = \exp(\Lambda \cdot t) \cdot w_{t0}$

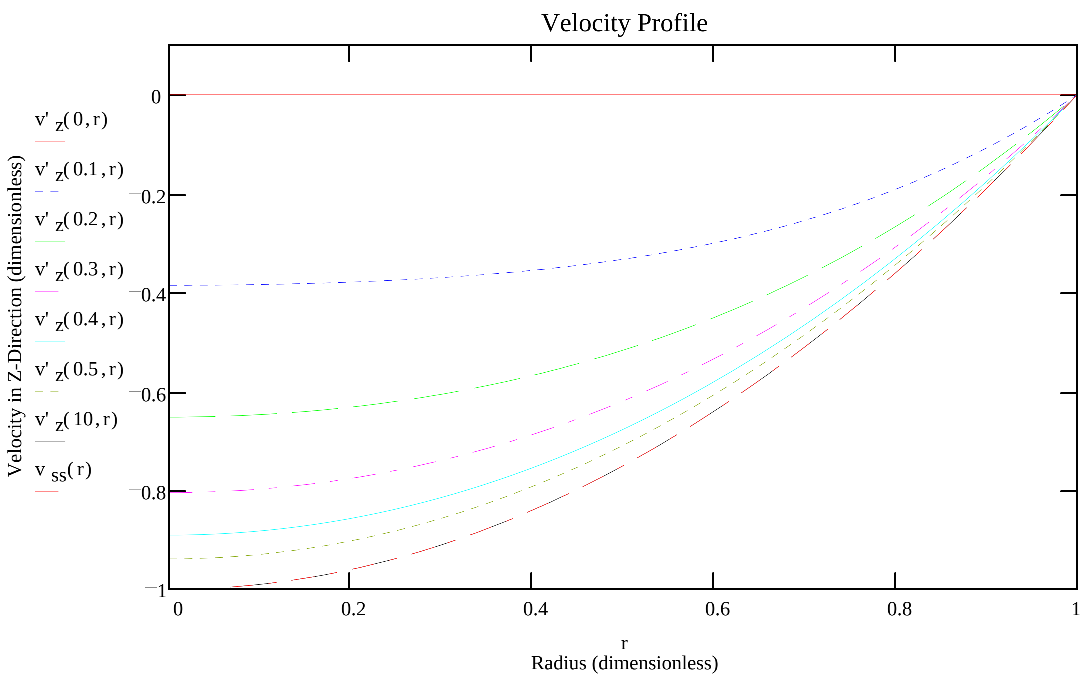
solution /w non-orthogonal basis vectors $\longrightarrow$ $w = \exp(A \cdot t) \cdot w'_{t0} = \left(V \cdot \exp(\Lambda \cdot t) \cdot V^{-1}\right) \cdot (J' \cdot A')$

$$\mathrm{expAt}(t) := V\cdot \mathrm{diag}\left(\overrightarrow{\exp\left(-\lambda_{power}{}^{2}\cdot t\right)}\right)\cdot V^{-1}$$

$$w'(t,r) := \mathrm{expAt}(t)\cdot \begin{bmatrix} 1 \\ r \\ r^2 \\ r^3 \\ r^4 \\ r^5 \\ r^6 \\ r^7 \\ r^8 \\ r^9 \\ r^{10} \end{bmatrix}\cdot A'$$

$$w'(t,r) := \sum_{j=0}^{n} \sum \overrightarrow{\left(\mathrm{expAt}(t)^{\langle j\rangle}\cdot A'\right)}\cdot r^{j}$$

$$v'_z(t,r) := w'(t,r) + v_{ss}(r)$$



- t=0 (initial condition)
- t=0.1
- t=0.2
- t=0.3
- t=0.4
- t=0.5
- t=10 (practically steady-state)
- Steady-state

Side Note. Express $w_{t0}(r)$ with non-orthogonal randomly generated Bessel series basis vectors.

Example of change-of-basis (example of similarity transform) $\quad bessel' = a \cdot bessel$

$$\begin{bmatrix} J'_0 \\ J'_1 \\ \dots \\ J'_n \end{bmatrix} = \begin{bmatrix} a_{00} & a_{01} & \dots & a_{0n} \\ a_{10} & a_{11} & \dots & a_{1n} \\ \dots & \dots & \dots & \dots \\ a_{n0} & a_{n1} & \dots & a_{nn} \end{bmatrix} \cdot \begin{bmatrix} J0(\lambda_0 \cdot r) \\ J0(\lambda_1 \cdot r) \\ \dots \\ J0(\lambda_n \cdot r) \end{bmatrix}$$

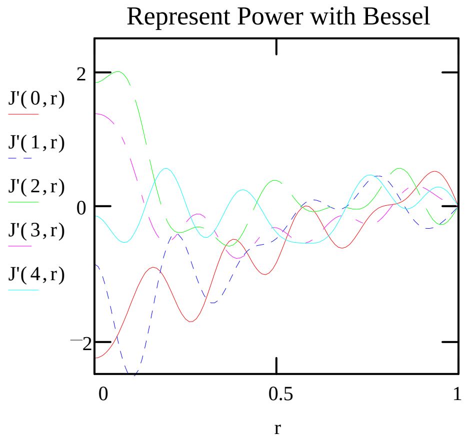
Randomly generate a second set of basis that is a linear combination of the original set of Bessel basis

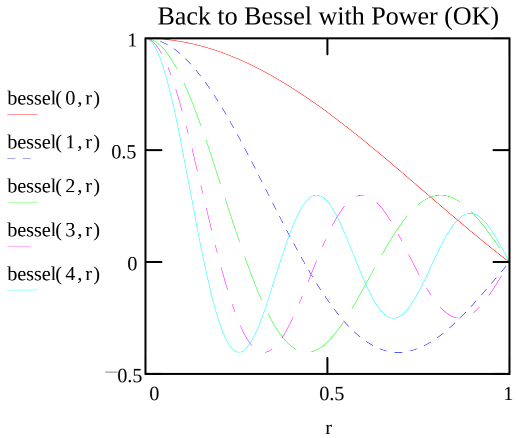
$$a_{i,j} := rnd(2) - 1$$

Unlike the previous non-example between power series and Bessel series, this is a change-of-basis (similarity transform) problem.

$$b := a^{-1}$$

$$J'(i,r) := \sum_{j=0}^{n} a_{i,j} \cdot J0(\lambda_j \cdot r) \qquad bessel(i,r) := \sum_{j=0}^{n} b_{i,j} \cdot J'(j,r)$$





Representation of w_{t0} with the second random power series -- coefficients to change from power series J to Bessel series

$$\begin{bmatrix} (w_{t0}(r), J'_0) \\ (w_{t0}(r), J'_1) \\ \dots \\ (w_{t0}(r), J'_n) \end{bmatrix} = \begin{bmatrix} (J'_0, J'_0) & (J'_1, J'_0) & \dots & (J'_n, J'_0) \\ (J'_0, J'_1) & (J'_1, J'_1) & \dots & (J'_n, J'_1) \\ \dots & \dots & \dots & \dots \\ (J'_0, J'_n) & (J'_1, J'_n) & \dots & (J'_n, J'_n) \end{bmatrix} \cdot \begin{bmatrix} A'_0 \\ A'_1 \\ \dots \\ A'_n \end{bmatrix}$$

$$wJ' = J'J'^{T} \cdot A' \qquad A' = J'J'^{-1} \cdot wJ'$$

scalar products of w_{t0} and basis J' matrix of scalar products of basis J'

$$wJ'_i := \int_0^1 r \cdot w_{t0}(r) \cdot J'(i,r)\, dr \qquad J'J'_{i,j} := \int_0^1 r \cdot J'(i,r) \cdot J'(j,r)\, dr$$

$$A' := J'J'^{-1} \cdot wJ' \qquad w'_{t0.approx}(r) := \sum_{i=0}^{n} A'_i \cdot J'(i,r)$$

The above calculates the coefficients to represent w_{t0} with power series Another way of calculating A' is based on the change-of-basis matrix a that relates A to A' that approximates a given vector|function w_{t0}

Representation of w_{t0} with Bessel basis

$$w_{t0.apprix} = \begin{pmatrix} J_0 & J_1 & \cdots & J_n \end{pmatrix} \cdot \begin{bmatrix} A_0 \\ A_1 \\ \cdots \\ A_n \end{bmatrix} = J \cdot A$$

Change-of-basis results in a different set of coefficients A' to describe the same vector|function w_{t0}

$$w'_{t0.apprix} = \begin{pmatrix} J'_0 & J'_1 & \cdots & J'_n \end{pmatrix} \cdot \begin{bmatrix} A'_0 \\ A'_1 \\ \cdots \\ A'_n \end{bmatrix} = \begin{pmatrix} J_0 & J_1 & \cdots & J_n \end{pmatrix} \cdot \begin{bmatrix} a_{00} & a_{10} & \cdots & a_{n0} \\ a_{01} & a_{11} & \cdots & a_{n1} \\ \cdots & \cdots & \cdots & \cdots \\ a_{0n} & a_{1n} & \cdots & a_{nn} \end{bmatrix} \cdot \begin{bmatrix} A'_0 \\ A'_1 \\ \cdots \\ A'_n \end{bmatrix} = J \cdot a^T \cdot A'$$

$$w'_{t0.apprix} = J' \cdot A' = J \cdot a^T \cdot A'$$

Thus, the two sets of coefficients A & A' are related by the change-of-basis matrix a

$$\begin{bmatrix} A_0 \\ A_1 \\ \cdots \\ A_n \end{bmatrix} = \begin{bmatrix} a_{00} & a_{10} & \cdots & a_{n0} \\ a_{01} & a_{11} & \cdots & a_{n1} \\ \cdots & \cdots & \cdots & \cdots \\ a_{0n} & a_{1n} & \cdots & a_{nn} \end{bmatrix} \cdot \begin{bmatrix} A'_0 \\ A'_1 \\ \cdots \\ A'_n \end{bmatrix}$$

$A = a^T \cdot A'$ ... A & A' in a column format

$A' = \left(a^T\right)^{-1} \cdot A$

Each method yields a different set of coefficients A to represent w_{t0} with Bessel series.

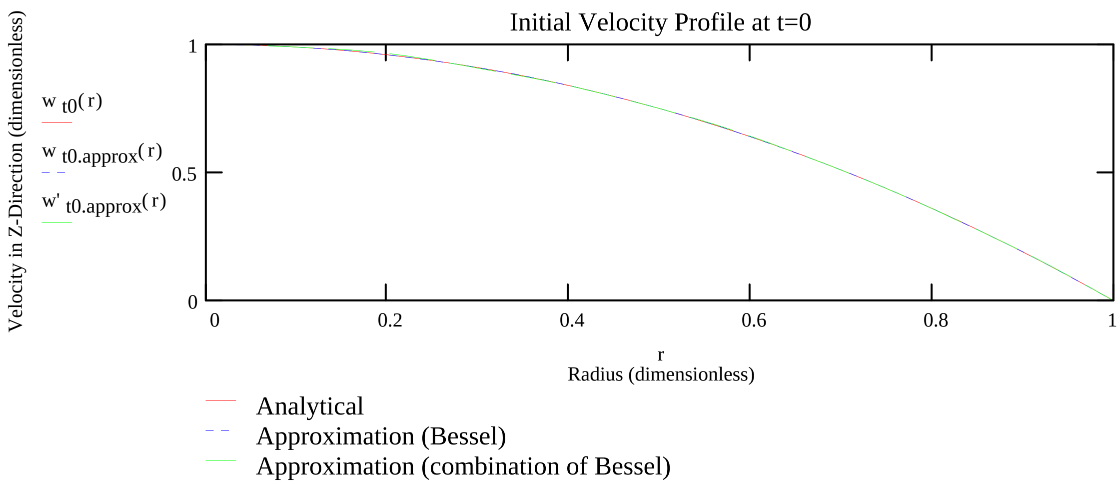
vector projection change-of-basis

A =

	0
0	1.108
1	-0.14
2	0.045
3	-0.021
4	0.012
5	-0.007
6	0.005
7	-0.003
8	0.003
9	-0.003
10	0.002

$a^T \cdot A'$ =

	0
0	1.108
1	-0.139
2	0.047
3	-0.018
4	0.011
5	-0.012
6	0.004
7	-0.005
8	0.003
9	-0.002
10	0.006



Representation of the *same* vector w with different set of basis J and J'.

$$w = \xi_0 \cdot J_0 + \xi_1 \cdot J_1 + \ldots + \xi_n \cdot J_n = \begin{pmatrix} J_0 & J_1 & \ldots & J_n \end{pmatrix} \cdot \begin{bmatrix} \xi_0 \\ \xi_1 \\ \ldots \\ \xi_n \end{bmatrix}$$

$$w = \xi'_0 \cdot J'_0 + \xi'_1 \cdot J'_1 + \ldots + \xi'_n \cdot J'_n = \begin{pmatrix} J'_0 & J'_1 & \ldots & J'_n \end{pmatrix} \cdot \begin{bmatrix} \xi'_0 \\ \xi'_1 \\ \ldots \\ \xi'_n \end{bmatrix}$$

Representation of the *same* output vector $\mathcal{L}\cdot w$ resulting from linear transformation $\mathcal{L}$ with different set of basis J and J'.

$$\mathcal{L}\cdot w = \xi_0\cdot\mathcal{L}\cdot J_0 + \xi_1\cdot\mathcal{L}\cdot J_1 + \ldots + \xi_n\cdot\mathcal{L}\cdot J_n = \xi_0\cdot\lambda_0\cdot J_0 + \xi_1\cdot\lambda_1\cdot J_1 + \ldots + \xi_n\cdot\lambda_n\cdot J_n$$

$$= \begin{pmatrix} J_0 & J_1 & \ldots & J_n \end{pmatrix}\cdot\begin{bmatrix} \lambda_0 & 0 & \ldots & 0 \\ 0 & \lambda_2 & \ldots & 0 \\ \ldots & \ldots & \ldots & \ldots \\ 0 & 0 & \ldots & \lambda_n \end{bmatrix}\cdot\begin{bmatrix} \xi_0 \\ \xi_1 \\ \ldots \\ \xi_n \end{bmatrix} = J\cdot\Lambda\cdot\xi$$

$$\mathcal{L}\cdot w = \xi'_0\cdot\mathcal{L}\cdot J'_0 + \xi'_1\cdot\mathcal{L}\cdot J'_1 + \ldots + \xi'_n\cdot\mathcal{L}\cdot J'_n$$

$$= \xi'_0\cdot\mathcal{L}\cdot\left(a_{00}\cdot J_0 + a_{01}\cdot J_1 + \ldots + a_{0n}\cdot J_n\right) + \xi'_1\cdot\mathcal{L}\cdot\left(a_{10}\cdot J_0 + a_{11}\cdot J_1 + \ldots + a_{1n}\cdot J_n\right) \ldots$$
$$+ \xi'_n\cdot\mathcal{L}\cdot\left(a_{n0}\cdot J_0 + a_{n1}\cdot J_1 + \ldots + a_{nn}\cdot J_n\right)$$

$$= \xi'_0\cdot\left(a_{00}\cdot\lambda_0\cdot J_0 + a_{01}\cdot\lambda_1\cdot J_1 + \ldots + a_{0n}\cdot\lambda_n\cdot J_n\right) + \xi'_1\cdot\left(a_{10}\cdot\lambda_0\cdot J_0 + a_{11}\cdot\lambda_1\cdot J_1 + \ldots + a_{1n}\cdot\lambda_n\cdot J_n\right) \ldots$$
$$+ \xi'_n\cdot\left(a_{n0}\cdot\lambda_0\cdot J_0 + a_{n1}\cdot\lambda_1\cdot J_1 + \ldots + a_{nn}\cdot\lambda_n\cdot J_n\right)$$

$$= \begin{pmatrix} J_0 & J_1 & \ldots & J_n \end{pmatrix}\cdot\begin{bmatrix} \lambda_0 & 0 & \ldots & 0 \\ 0 & \lambda_1 & \ldots & 0 \\ \ldots & \ldots & \ldots & \ldots \\ 0 & 0 & \ldots & \lambda_n \end{bmatrix}\cdot\begin{bmatrix} a_{00} & a_{10} & \cdots & a_{n0} \\ a_{01} & a_{11} & \cdots & a_{n1} \\ \ldots & \ldots & \ldots & \ldots \\ a_{0n} & a_{1n} & \cdots & a_{nn} \end{bmatrix}\cdot\begin{bmatrix} \xi_0 \\ \xi_1 \\ \ldots \\ \xi_n \end{bmatrix} = J\cdot\Lambda\cdot a^T\cdot\xi$$

We can define the linear transformation $\mathcal{L}$ based on what it does to each of the basis vectors J

$$\mathcal{L}\cdot J = \mathcal{L}\cdot\begin{pmatrix} J_0 & J_1 & \ldots & J_n \end{pmatrix} = \begin{pmatrix} \mathcal{L}\cdot J_0 & \mathcal{L}\cdot J_1 & \ldots & \mathcal{L}\cdot J_n \end{pmatrix} = \begin{pmatrix} \lambda_0\cdot J_0 & \lambda_1\cdot J_1 & \ldots & \lambda_n\cdot J_n \end{pmatrix}$$

$$= \begin{pmatrix} J_0 & J_1 & \ldots & J_n \end{pmatrix}\cdot\begin{bmatrix} \lambda_0 & 0 & \ldots & 0 \\ 0 & \lambda_1 & \ldots & 0 \\ \ldots & \ldots & \ldots & \ldots \\ 0 & 0 & \ldots & \lambda_n \end{bmatrix} = J\cdot\Lambda$$

We can define the same linear transformation $\mathcal{L}$ equally well based on what it does to any set of basis vectors J'

$$\mathcal{L}\cdot J' = \mathcal{L}\cdot\begin{pmatrix} J'_0 & J'_1 & \ldots & J'_n \end{pmatrix} = \begin{pmatrix} \mathcal{L}\cdot J'_0 & \mathcal{L}\cdot J'_1 & \ldots & \mathcal{L}\cdot J'_n \end{pmatrix}$$

$$= \left[\mathcal{L}\cdot\left(a_{00}\cdot J_0 + a_{01}\cdot J_1 + \ldots + a_{0n}\cdot J_n\right) \quad \mathcal{L}\cdot\left(a_{10}\cdot J_0 + a_{11}\cdot J_1 + \ldots + a_{1n}\cdot J_n\right) \quad \ldots \quad \mathcal{L}\cdot\left(a_{n0}\cdot J_0 + a_{n1}\cdot J_1 + \ldots + a_{nn}\cdot J_n\right)\right]$$

$$= \left[\left(a_{00}\cdot\lambda_0\cdot J_0 + a_{01}\cdot\lambda_1\cdot J_1 + \ldots + a_{0n}\cdot\lambda_n\cdot J_n\right) \quad \mathcal{L}\cdot\left(a_{10}\cdot\lambda_0\cdot J_0 + a_{11}\cdot\lambda_1\cdot J_1 + \ldots + a_{1n}\cdot\lambda_n\cdot J_n\right) \quad \ldots \quad \mathcal{L}\cdot\left(a_{n0}\cdot\lambda_0\cdot J_0 + a_{n1}\right.\right.$$

$$= \begin{pmatrix} J_0 & J_1 & \ldots & J_n \end{pmatrix}\cdot\begin{bmatrix} \lambda_0 & 0 & \ldots & 0 \\ 0 & \lambda_1 & \ldots & 0 \\ \ldots & \ldots & \ldots & \ldots \\ 0 & 0 & \ldots & \lambda_n \end{bmatrix}\cdot\begin{bmatrix} a_{00} & a_{10} & \cdots & a_{n0} \\ a_{01} & a_{11} & \cdots & a_{n1} \\ \ldots & \ldots & \ldots & \ldots \\ a_{0n} & a_{1n} & \cdots & a_{nn} \end{bmatrix} = J\cdot\Lambda\cdot a^T$$

or, simply substitute in $J' = a \cdot J$ we obtain: $\mathcal{L} \cdot J' = \mathcal{L} \cdot \left(J \cdot a^T\right) = (\mathcal{L} \cdot J) \cdot a^T = (J \cdot \Lambda) \cdot a^T = J \cdot \Lambda \cdot a^T$

Summary: action of linear transformation $\mathcal{L}$ on basis vectors J or J'

J & J' in row format

$J' = J \cdot a^T$

$\mathcal{L} \cdot J = J \cdot \Lambda$

$\mathcal{L} \cdot J' = J \cdot \Lambda \cdot a^T = \left[J' \cdot \left(a^T\right)^{-1}\right] \cdot \Lambda \cdot a^T = J' \cdot \left[\left(a^T\right)^{-1} \cdot \Lambda \cdot a^T\right]$

J & J' in column format

$J' = a \cdot J$

$\mathcal{L} \cdot J = \Lambda \cdot J$

$\mathcal{L} \cdot J' = a \cdot \Lambda \cdot J = a \cdot \Lambda \cdot \left(a^{-1} \cdot J'\right) = \left(a \cdot \Lambda \cdot a^{-1}\right) \cdot J'$

Matrices in similarity transform (vectors in a column format)

compare definition of a matrix L to describe the action of $\mathcal{L}$. Compare it to the line above

$\mathcal{L} \cdot J' = L \cdot J'$ ⟵ compare ⟶ $\mathcal{L} \cdot J' = \left(a \cdot \Lambda \cdot a^{-1}\right) \cdot J'$ ⟶ $\Lambda_{i,i} := \lambda_i$ $L := a \cdot \Lambda \cdot a^{-1}$

We see that matrix "L" in this problem corresponds to the matrix "A" in the "standard" notation. Likewise, matrix "a" in this problem corresponds to the transform matrix "V" in the "standard" notation.

$L \cdot a = a \cdot \Lambda$ ... notation in this problem

$A \cdot V = V \cdot \Lambda$ ... "standard" notation in similarity transform

Eigenvalue-eigenvector for the linear transform $\mathcal{L}_{\mathcal{R}}$: $\lambda_{J'} := \mathrm{eigenvals}(L)$ $V := \mathrm{eigenvecs}(L)$

Note that Mathcad returns $\lambda_{J'}$ that is arranged differently from λ. Likewise, V is arranged differently from a.

Rearrange in ascending order $\lambda V := \mathrm{augment}\left(\lambda_{J'}, V^T\right)$ $\lambda V := \mathrm{csort}(\lambda V, 0)$

$\lambda_{J'} := \mathrm{submatrix}(\lambda V, 0, \mathrm{rows}(\lambda V) - 1, 0, 0)$ $\Lambda_{J'} := \mathrm{diag}\left(\lambda_{J'}\right)$

$V := \mathrm{submatrix}(\lambda V, 0, \mathrm{rows}(\lambda V) - 1, 1, \mathrm{cols}(\lambda V) - 1)^T$

check: $\lambda^T =$

	0	1	2	3	4	5	6	7	8	9	10
0	2.405	5.52	8.654	11.792	14.931	18.071	21.212	24.352	27.493	30.635	33.776

$\lambda_{J'}^T =$

	0	1	2	3	4	5	6	7	8	9	10
0	2.405	5.52	8.654	11.792	14.931	18.071	21.212	24.352	27.493	30.635	33.776

Mathcad returns normalized eigenvectors V. To better compare matrix "a" with matrix "V", normalize "a".

$$a_{norm}^{\langle i \rangle} := \frac{a^{\langle i \rangle}}{\sqrt{\left(a^T \cdot a\right)_{i,i}}}$$

check: Note that a_{norm}=V is not orthogonal, and some eigenvectors differ in sign.

V =

	0	1	2	3	4
0	0.523	0.333	0.097	0.194	0.346
1	0.4	0.534	0.036	-0.132	-0.358
2	-0.041	0.041	0.415	-0.362	0.533
3	0.235	-0.096	0.387	0.019	0.261
4	0.079	-0.019	0.288	0.428	-0.009
5	0.364	-0.35	-0.354	-0.41	-0.369
6	-0.018	0.08	0.515	-0.064	-0.03
7	$-9.571\cdot10^{-4}$	0.514	0.083	-0.041	0.368
8	-0.351	-0.109	-0.284	0.645	0.328
9	-0.206	0.069	0.089	-0.166	0.004
10	-0.453	-0.429	-0.313	0.115	0.137

a_{norm} =

	0	1	2	3	4
0	-0.523	-0.333	0.097	-0.194	0.346
1	-0.4	-0.534	0.036	0.132	-0.358
2	0.041	-0.041	0.415	0.362	0.533
3	-0.235	0.096	0.387	-0.019	0.261
4	-0.079	0.019	0.288	-0.428	-0.009
5	-0.364	0.35	-0.354	0.41	-0.369
6	0.018	-0.08	0.515	0.064	-0.03
7	$9.571\cdot10^{-4}$	-0.514	0.083	0.041	0.368
8	0.351	0.109	-0.284	-0.645	0.328
9	0.206	-0.069	0.089	0.166	0.004
10	0.453	0.429	-0.313	-0.115	0.137

Dynamics with non-orthogonal basis vs orthogonal basis

Analogy with 1st-order ODEs $\frac{d}{dt}z = \Lambda\cdot z$ solution $\longrightarrow$ $z = \exp(\Lambda\cdot t)\cdot z_0$

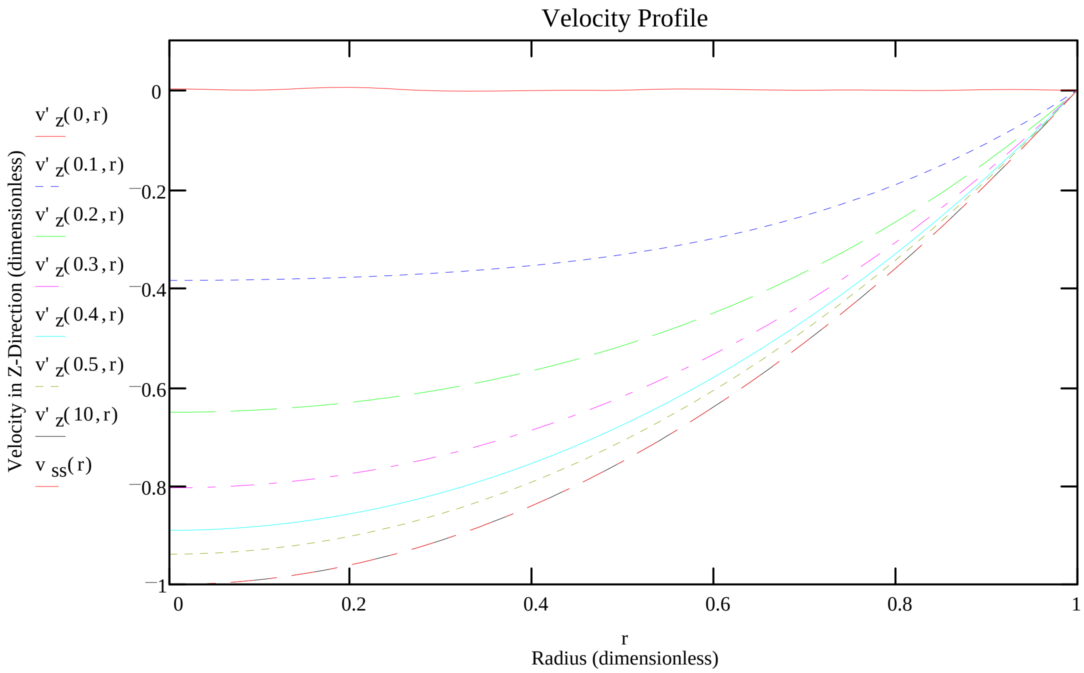
For PDE in this problem $\frac{d}{dt}w = \mathcal{L}_{\mathcal{R}}w$ solution $\longrightarrow$ $w = \exp(\Lambda\cdot t)\cdot w_{t0}$

solution /w non-orthogonal basis vectors $\longrightarrow$ $w = \exp(A\cdot t)\cdot w'_{t0} = \left(V\cdot\exp(\Lambda\cdot t)\cdot V^{-1}\right)\cdot(J'\cdot A')$

$$\mathrm{expAt}(t) := V\cdot \mathrm{diag}\left(\overrightarrow{\exp\left(-\lambda_{J'}^{2}\cdot t\right)}\right)\cdot V^{-1}$$

$$w'(t,r) := \mathrm{expAt}(t)\cdot\begin{bmatrix} J'(0,r)\\ J'(1,r)\\ J'(2,r)\\ J'(3,r)\\ J'(4,r)\\ J'(5,r)\\ J'(6,r)\\ J'(7,r)\\ J'(8,r)\\ J'(9,r)\\ J'(10,r)\end{bmatrix}\cdot A' \qquad w'(t,r) := \sum_{j=0}^{n}\sum\overrightarrow{\left(\mathrm{expAt}(t)^{\langle j\rangle}\cdot A'\right)}\cdot J'(j,r)$$

$$v'_z(t,r) := w'(t,r) + v_{ss}(r)$$



t=0 (initial condition)
t=0.1
t=0.2
t=0.3
t=0.4
t=0.5
t=10 (practically steady-state)
Steady-state

Side Note. Express $w_{t0}(r)$ with non-orthogonal Bessel series basis vectors (due to a different scalar product definition with a different weight function) ., although the basis remain unchanged. This is not an example of change-of-basis.

Representation of w_{t0} with the Bessel series

$$\begin{bmatrix} (w_{t0}(r), J_0) \\ (w_{t0}(r), J_1) \\ \dots \\ (w_{t0}(r), J_n) \end{bmatrix} = \begin{bmatrix} (J_0, J_0) & (J_1, J_0) & \dots & (J_n, J_0) \\ (J_0, J_1) & (J_1, J_1) & \dots & (J_n, J_1) \\ \dots & \dots & \dots & \dots \\ (J_0, J_n) & (J_1, J_n) & \dots & (J_n, J_n) \end{bmatrix} \cdot \begin{bmatrix} A'_0 \\ A'_1 \\ \dots \\ A'_n \end{bmatrix}$$

$wJ' = J'J'^T \cdot A'$ $\quad A' = J'J'^{-1} \cdot wJ'$

scalar products of w_{t0} and basis J' matrix of scalar products of basis J'

$$wJ'_i := \int_0^1 w_{t0}(r) \cdot J0(\lambda_i \cdot r)\, dr \qquad J'J'_{i,j} := \int_0^1 J0(\lambda_i \cdot r) \cdot J0(\lambda_j \cdot r)\, dr$$ different weight function

$$A' := J'J'^{-1} \cdot wJ' \qquad w'_{t0.approx}(r) := \sum_{i=0}^{n} A'_i \cdot J0(\lambda_i \cdot r)$$

The above calculates the coefficients to represent w_{t0} with Bessel series, but with a different scalar product definition. A product definition that makes the basis non-orthogonal basically produces the same set of coefficients, albeit with more calculations from matrix inverse.

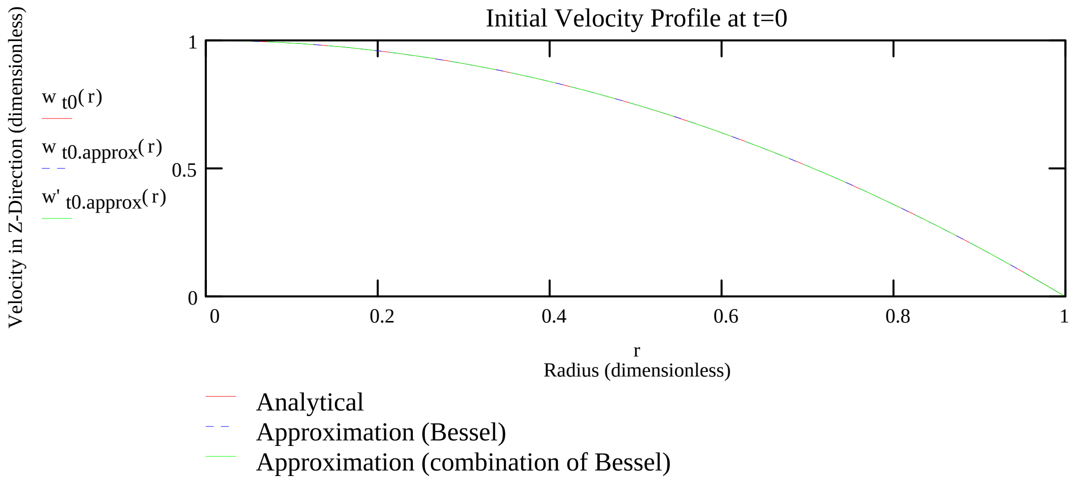
$A^T =$

	0	1	2	3	4	5	6	7	8	9	10
0	1.108	-0.14	0.045	-0.021	0.012	-0.007	0.005	-0.003	0.003	-0.003	0.002

$A'^T =$

	0	1	2	3	4	5	6	7	8	9	10
0	1.108	-0.14	0.045	-0.021	0.012	-0.007	0.005	-0.003	0.002	-0.002	0.001

Since the coefficients are the same, the initial velocity profile is approximated similarly. Because the exponential decay of each mode remains identical (this decay behavior is not affected by the scalar product definition), so does the subsequent velocity profile. This non-orthogonal example demonstrates that orthogonality only simplifies the calculation of the coefficients when we try to represent the initial velocity profile, but does not affect the evolution of solution with time. The eigenvectors from the $\mathcal{L}_{\mathcal{T}}$ operator are not orthogonal, nor does orthogonality enters into the time-dependent behavior. The coupling of the $\mathcal{L}_{\mathcal{T}}$ operator and the $\mathcal{L}_{\mathcal{R}}$ operator in the original given PDE dictates how pairs of eigenvectors from these two operators (T_i and R_i) are coupled to form overall eigenvectors $wi = T_i \cdot R_i$. This coupling determines how each mode (each eigenvector component) evolves with time, independent of all the other modes. And this coupling does not depend on the definition of a scalar product, where orthogonality becomes affected.



The following formula remain unchanged; orthogonality does not enter into the dynamics part of the solution.

$$w'(t,r) := \sum_{i=0}^{n} A'_i \cdot \exp\left[-\left(\lambda_i\right)^2 \cdot t\right] \cdot J0\left(\lambda_i \cdot r\right)$$

Finally, the dimensionless z-direction velocity is: $v'_z(t,r) := w'(t,r) + v_{ss}(r)$