NRCExecSec Resource

From: Timothy Margulies <t.margulies@gmail.com>
Sent: Thursday, June 14, 2018 11:06 AM
To: Allan Pierce; amb18; ABDUL M SIDDIQUI; Henry Bart; chickey@olemiss.edu; jared cohon; David T Blackstock; Eagles.tom@epa.gov; evishni1@jhu.edu; gadelhak@vcu.edu; Hamilton, Mark F; hora@usc.edu; ilenebv@beogrinconsulting.com; korman@usna.edu; kstebe@seas.upenn.edu; Katherine J. Kuchenbecker; Marc Donohue; pk.niyogi@hq.doe.gov; prosperetti@jhu.edu; Puskin.jerome@epa.gov; raspet@olemiss.edu; rlpowell; sdavis@northwestern.edu; shenton@udel.edu; Brian Spooner; staehle@rowan.edu; Vladimir.Genis@drexel.edu; Weinstock.larry@epa.gov; william.j.danley@drexel.edu; NRCExecSec Resource; Hattaway, Amanda
Subject: [External_Sender] Learning thru MAPLE models-beginners to advanced researchers
Attachments: MapleappsTiM.pdf

About the Report Collection

Growing up with the "Sputnik generation" that needed to catch-up in math and the sciences following the satellite orbiting demonstration, industry aiming for automation and digital systems, as well as, understanding environmental pollution has significantly influenced societies' education and jobs. The computer concept attributed to von Neumann of a *programmable calculator*, such as, an electronic numerical integrator has advanced an ambitious craze with accompanying technological innovations.

This purpose of this report is to provide a variety of applied math problems with computation on selected topics of physics, engineering, and anthropology. Readers are encouraged to get their start on their own journey of math explorations with computation with experiential learning.

Additional mostly analytical studies are published in,

Mathematical Science Explorations, Create Space Pub. ISBN-13: 978-1986909952 (2018).

Timothy S. Margulies

Math & Computations: MAPLE Explorations

T. Margulies

About the Report Collection

Growing up with the "Sputnik generation" that needed to catch-up in math and the sciences following the 1957 satellite orbiting demonstration, industry aiming for automation and digital systems, as well as, understanding environmental pollution has significantly influenced societies' education and jobs. The computer concept attributed to von Neumann of a *programmable calculator*, such as, an electronic numerical integrator as advanced with technological innovations.

This purpose of this report is to provide a variety of applied math problems with computation on selected topics of physics, engineering, and anthropology. Readers are encouraged to get their cravings and start on their own journey of math explorations with computation with experiential learning.

About The Author

Timothy S. Margulies enjoys presenting a collection of investigations on the computer for introducing and for teaching quantitative studies, to motivate students to conduct their own explorations on neoclassical engineering physics onto validations with data analysis. He enjoys applied mathematical recreations, as well as, photography, art, and bicycling. He earned both Masters and Doctorate degrees at The Johns Hopkins University in Baltimore, Maryland.

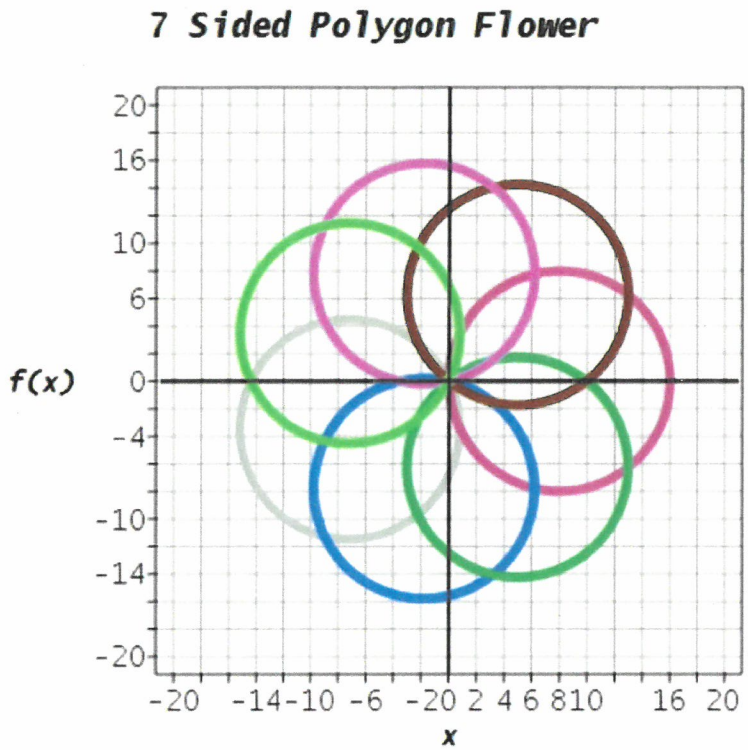
	Topic	Page
1.1	Circle of Life & Polygons	4
1.2	Logarithm & Exponential Functions	5
1.3	Geometry with Lines	6
1.4	Golden Parabola with Roots, Functions, Riemann Sums	7
1.5	Polynomials	8
1.6	City Grid Design	9
1.7	Ancient Cities-Monuments	10
1.8	Parabola and Tangent Line at a Point	11
1.9	Functions, Differentials, Integrals, & Riemann Sums	12
2.0	Planetary Orbits	16
2.1	Bicycle Natural Frequencies	17
2.2	Linear Bicycle Dynamic System	18
2.3	Simple Tire Model Dimensionless Parameters	28
2.4	Nonlinear Spring-Pendulum Model	29
2.5	Scotch Crank	30
2.6	Volcanic Particles	31
2.7	Biodynamic Model	34
2.8	Flamant, Boussinesq Models	35
2.9	Fractional Calculus Bubble Oscillator	39
3.0	Epidemiological Model	42
3.1	Pi Theorem Parameters: Falling Sphere	46
3.2	Accident Contaminated Tuna Off West Coast	52
3.3	Egypt's Great Pyramid's Heights versus Orion Belt Star's Brightness	55
3.4	Time of Arrival Acoustic Emission Locator	56
3.5	Four Degrees of Freedom Motorcycle Dynamics Model	57
3.6	Interest Compounding, Present Value, Capital Recovery Factor calculations	58
3.7	Bar Chart	59
3.8	Eigenvalues, Eigenvector Diagonalization	60
3.9	Regional Nuclear Generating Risk Allocations	63
4.0	Spherical Trigonometry Haversine Distance Calculation	65

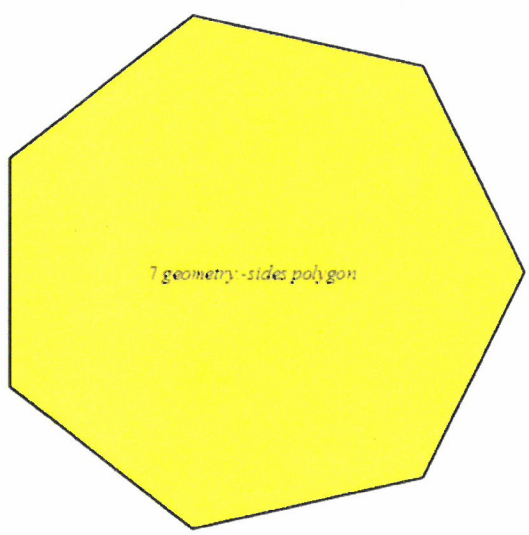
© 2018

by Timothy S. Margulies

Dedicated To: Parents, Children, educators, and the MAPLE user community

1.1 Circle of Life & Polygons

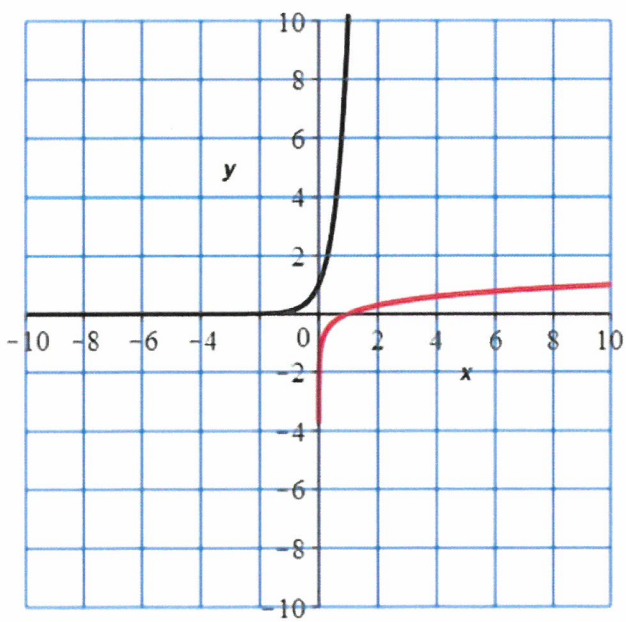




restart; with(plots); with(Student); with(RandomTools); with(geometry); n1 := 7; r1 := n1+1; x1 := 3*n1; col1 := seq(COLOR(RGB, rand()/10^12, rand()/10^12, rand()/10^12), j = 1 .. n1); theta1 :=

seq(i*(2*3.14159)/n1, I = 1 .. n1); `cosθ1` := seq(cos(theta1[i]), I = 1 .. n1); `sinθ1` := seq(sin(theta1[i]), I = 1 .. n1); for I to n1 do h1[i] := r1*`cosθ1`[i]; k1[i] := r1*`sinθ1`[i]; pl[i] := implicitplot((x-h1[i])^2+(y-k1[i])^2-r1^2, x = -x1 .. x1, y = -x1 .. x1, color = col1[i], thickness = 5, axes = boxed, title = 'n1*Sided*Polygon*Flower', titlefont = [COURIER, Bold, 14], labels = ["x", "f(x)"], labelfont = [COURIER, BOLDITALIC, 12], axesfont = [COURIER, 12], gridlines, tickmarks = [20, 20], scaling = constrained) end do; g := proc (x) options operator, arrow; 0*x end proc; h := proc (x) options operator, arrow; 10000000*x end proc; pl2 := plot(g(x), x = -x1 .. x1, y = -x1 .. x1, color = black, thickness = 2, axes = boxed, title = 'n1*Sided*Polygon*Flower', titlefont = [COURIER, Bold, 14], labels = ["x", "f(x)"], labelfont = [COURIER, BOLDITALIC, 12], axesfont = [COURIER, 12], gridlines, tickmarks = [10, 10], scaling = constrained); pl3 := plot(h(x), x = -x1 .. x1, y = -x1 .. x1, color = black, thickness = 1, axes = boxed, title = 'n1*Sided*Polygon*Flower', titlefont = [COURIER, Bold, 14], labels = ["x", "f(x)"], labelfont = [COURIER, BOLDITALIC, 12], axesfont = [COURIER, 12], gridlines, tickmarks = [20, 20], scaling = constrained); display({pl2, pl3, seq(pl[j], j = 1 .. n1)}); RegularPolygon(pgon, n1, point(o, 0, 0), 16); detail(pgon); pgon2 := proc (n1) options operator, arrow; [seq([cos(2*Pi*j/n1), sin(2*Pi*j/n1)], j = 1 .. n1)] end proc; display([polygonplot(pgon2(n1), color = yellow), textplot([0, 0, n1*sides*polygon])], axes = none); g := proc (x) options operator, arrow; 0*x end proc; h := proc (x) options operator, arrow; 10000000*x end proc; plt2 := plot(g(x), x = -x1 .. x1, y = -x1 .. x1, color = black, thickness = 2, axes = boxed, gridlines, tickmarks = [10, 10], scaling = constrained); plt3 := plot(h(x), x = -x1 .. x1, y = -x1 .. x1, color = black, thickness = 1, axes = boxed, gridlines, tickmarks = [20, 20], scaling = constrained); display({plt2, plt3});

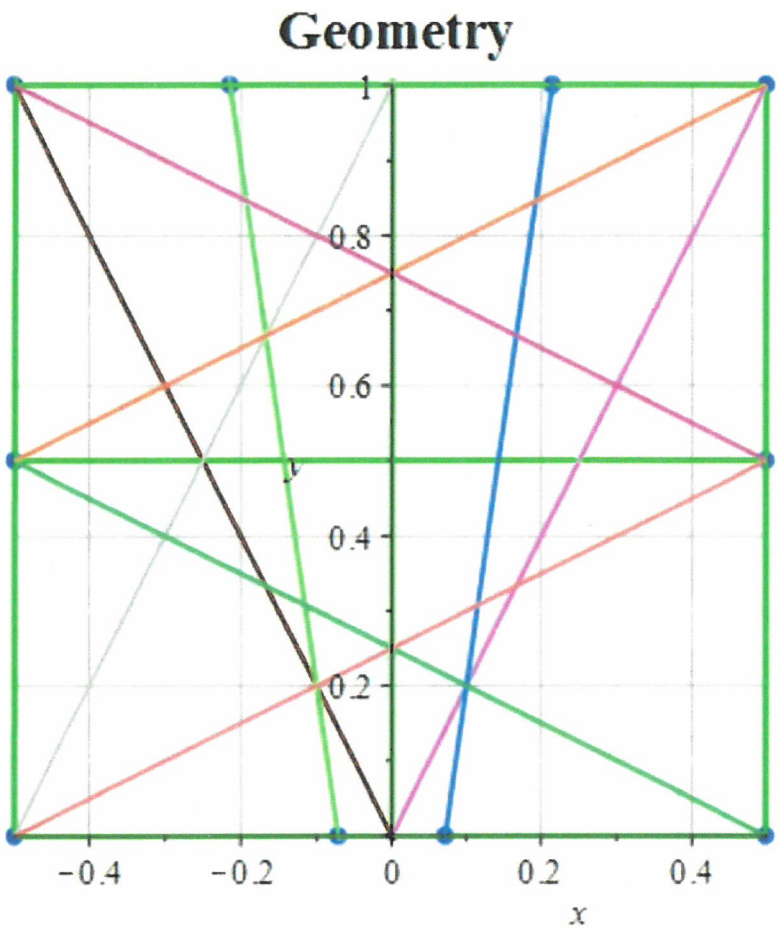
1.2 Logarithm & Exponential Functions



restart; with(Student); with(plots); p1 := plot(0*x, x = -10 .. 10, y = -10 .. 10, axis = [gridlines = [10, color = blue]], labelfont = [COURIER, BOLDITALIC, 12], axesfont = [BOLDCOURIER, 14], gridlines, tickmarks = [10, 10]); p2 := plot(x/10^9, x = -10 .. 10, y = -10 .. 10, axis = [gridlines = [10, color = blue]], labelfont =

[COURIER, BOLDITALIC, 12], axesfont = [BOLDCOURIER, 14], gridlines, tickmarks = [10, 10]); f := proc (x) options operator, arrow; 10^x end proc; p3 := plot(f(x), x = -10 .. 10, y = -10 .. 10, axis = [gridlines = [10, color = blue]], labelfont = [COURIER, BOLDITALIC, 12], axesfont = [BOLDCOURIER, 14], gridlines, tickmarks = [10, 10], color = black, thickness = 3); g := proc (x) options operator, arrow; log10(x) end proc; p4 := plot(g(x), x = -10 .. 10, y = -10 .. 10, axis = [gridlines = [10, color = blue]], labelfont = [COURIER, BOLDITALIC, 12], axesfont = [BOLDCOURIER, 14], gridlines, tickmarks = [10, 10], color = red, thickness = 3); display({p1, p2, p3, p4});

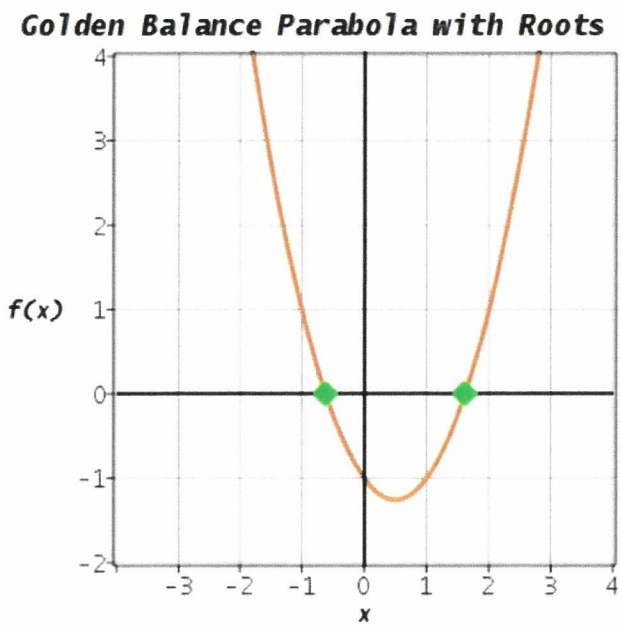
1.3 Geometry with Lines



restart; with(Student); with(plots); with(RandomTools); n := 10; col1 := [seq(COLOR(RGB, rand()/10^12, rand()/10^12, rand()/10^12), j = 1 .. n)]; a := 1; p0 := [0, a]; p1 := [-(1/2)*a, a]; p2 := [-3/14*a, a]; p3 := [3/14*a, a]; p4 := [(1/2)*a, a]; p5 := [(1/2)*a, (1/2)*a]; p6 := [(1/2)*a, 0]; p7 := [-(1/2)*a, 0]; p8 := [-(1/2)*a, (1/2)*a]; q1 := [-(1/14)*a, 0]; q2 := [(1/14)*a, 0]; L1 := y = -(1/2)*x+(1/4)*a; L2 := y = (1/2)*x+(1/4)*a; L3 := y = -2*x; L4 := y = 2*x; L5 := y = -7*x-(1/2)*a; L6 := y = 7*x-(1/2)*a; L7 := y = -(1/2)*x+3*a*(1/4); L8 := y = (1/2)*x+3*a*(1/4); L9 := y = 2*x+a; L10 := y = 2*x+a; L1x := proc (x) options operator, arrow; -(1/2)*x+(1/4)*a end proc; L2x := proc (x) options operator, arrow; (1/2)*x+(1/4)*a end

proc; L3x := proc (x) options operator, arrow; -2*x end proc; L4x := proc (x) options operator, arrow; 2*x end proc; L5x := proc (x) options operator, arrow; -7*x-(1/2)*a end proc; L6x := proc (x) options operator, arrow; 7*x-(1/2)*a end proc; L7x := proc (x) options operator, arrow; -(1/2)*x+(3/4)*a end proc; L8x := proc (x) options operator, arrow; (1/2)*x+(3/4)*a end proc; L9x := proc (x) options operator, arrow; 2*x+a end proc; L10x := proc (x) options operator, arrow; 2*x+a end proc; Lup := proc (x) options operator, arrow; 0*x+a end proc; Llow := proc (x) options operator, arrow; 0*x end proc; Lc := proc (x) options operator, arrow; 1000000*x end proc; LcL := proc (x) options operator, arrow; 10000000*x+5000000*a end proc; LcR := proc (x) options operator, arrow; Heaviside(x-(1/2)*a) end proc; Lh := proc (x) options operator, arrow; 0*x+(1/2)*a end proc; p17 := solve({L5, L7}, {x, y}); p16 := solve({L5, L8}, {x, y}); p10 := solve({L1, L5}, {x, y}); p51 := solve({L2, L7}, {x, y}); outer := [p1, p2, p3, p4, p5, p6, q1, q2, p7, p8]; plpts := pointplot(outer, style = point, symbol = solidcircle, symbolsize = 14, color = blue, axes = normal, gridlines, title = "Geometry", titlefont = ["ARIAL", bold, 18], scaling = constrained); x1 := -.5; x2 := .5; x21 := .500000000001; y1 := 0.; y2 := 1.; plinesxy := plot({Lc(x), LcL(x), LcR(x), Lh(x), Llow(x), Lup(x)}, x = x1 .. x21, y = y1 .. y2, axes = normal, gridlines = true, title = "Geometry ", titlefont = ["ARIAL", bold, 18], style = line, thickness = [2, 2], color = green); plines := plot({L10x(x), L1x(x), L2x(x), L3x(x), L4x(x), L5x(x), L6x(x), L7x(x), L8x(x), L9x(x)}, x = x1 .. x2, y = y1 .. y2, axes = normal, gridlines = true, title = "Geometry ", titlefont = ["ARIAL", bold, 18], style = line, thickness = [2, 2], color = col1); plots[display]({plines, plpts, plinesxy});

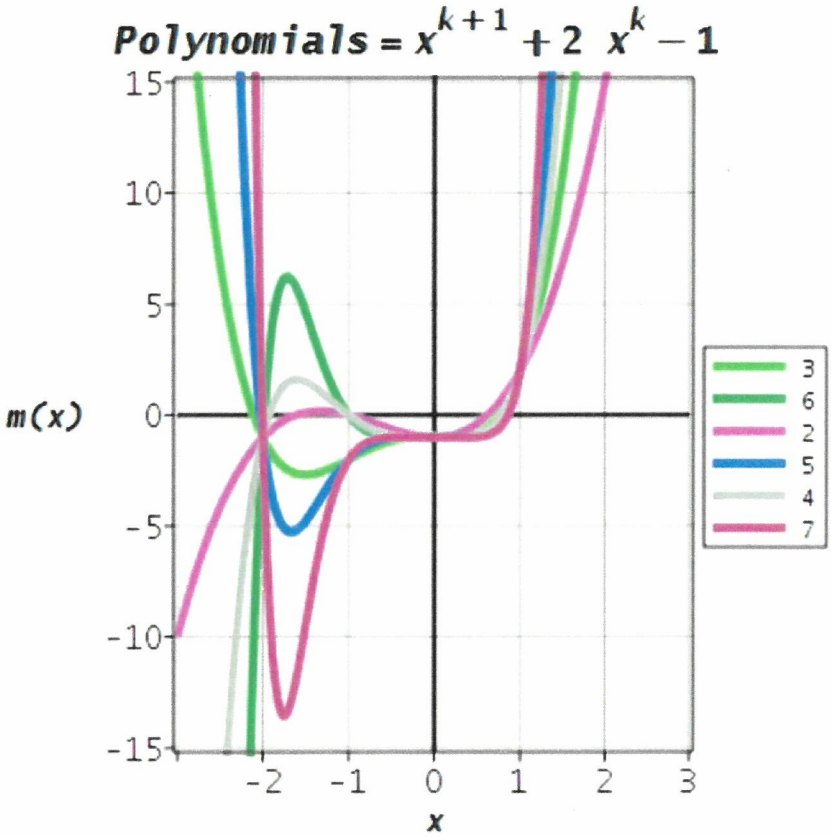
1.4 Golden Parabola with Roots



restart; with(plots); with(Student); with(RandomTools); f := proc (x) options operator, arrow; x^2-x-1 end proc; eq1 := f(x) = 0; sol := solve(eq1, x); r1 := evalf(sol[1]); r2 := evalf(sol[2]); g := proc (x) options operator, arrow; 0*x end proc; h := proc (x) options operator, arrow; 10000000*x end proc; p5 := plot([[r2, 0], [r1, 0]], x = -4 .. 4, y = -2 .. 4, style = point, symbol = soliddiamond, symbolsize = 30, color =

green, axes = boxed, title = 'Golden*Balance*Parabola*with*Roots', titlefont = [COURIER, Bold, 14], labels = ["x", "f(x)"], labelfont = [COURIER, BOLDITALIC, 12], axesfont = [COURIER, 12], gridlines, tickmarks = [10, 10]); p6 := plot(f(x), x = -4 .. 4, y = -2 .. 4, color = gold, thickness = 3, axes = boxed, title = 'Golden*Balance*Parabola*with*Roots', titlefont = [COURIER, Bold, 14], labels = ["x", "f(x)"], labelfont = [COURIER, BOLDITALIC, 12], axesfont = [COURIER, 12], gridlines, tickmarks = [10, 10]); p7 := plot(g(x), x = -4 .. 4, y = -2 .. 4, color = black, thickness = 2, axes = boxed, title = 'Golden*Balance*Parabola*with*Roots', titlefont = [COURIER, Bold, 14], labels = ["x", "f(x)"], labelfont = [COURIER, BOLDITALIC, 12], axesfont = [COURIER, 12], gridlines, tickmarks = [10, 10]); p8 := plot(h(x), x = -4 .. 4, y = -2 .. 4, color = black, thickness = 2, axes = boxed, title = 'Golden*Balance*Parabola*with*Roots', titlefont = [COURIER, Bold, 14], labels = ["x", "f(x)"], labelfont = [COURIER, BOLDITALIC, 12], axesfont = [COURIER, 12], gridlines, tickmarks = [10, 10]); display({p5, p6, p7, p8});

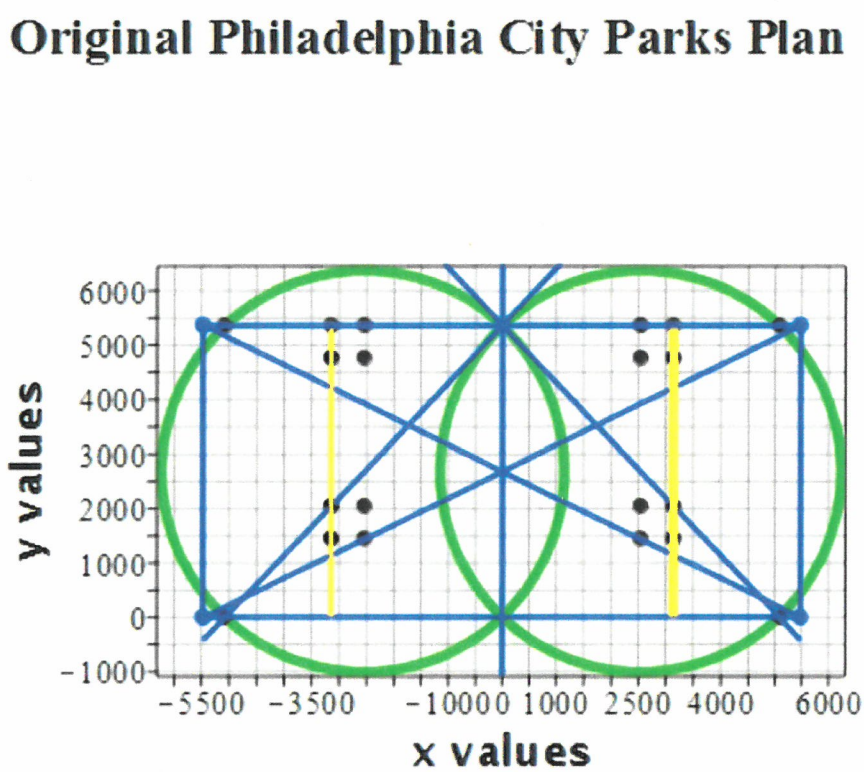
1.5 Polynomials



restart; with(plots); with(Student); with(RandomTools); n := 7; r00 := 3; r11 := 15; col1 := seq(COLOR(RGB, rand()/10^12, rand()/10^12, rand()/10^12), j = 1 .. n); for I from 2 to n do m := proc (x) options operator, arrow; x^(i+1)+2*x^i-1 end proc; eqn := m(x) = 0; sol := solve(eqn, x); rt :=

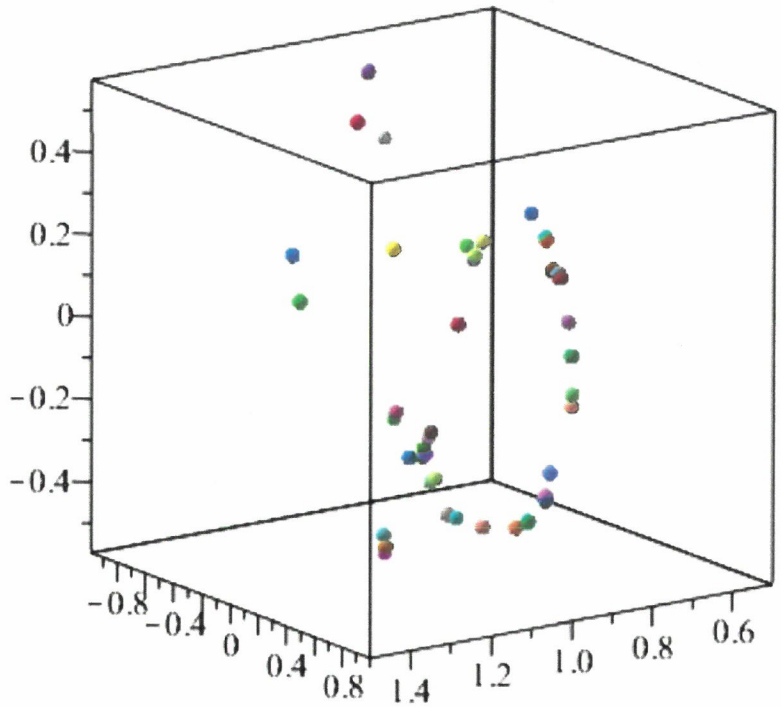
```
seq(evalf(Re(sol[j])), j = 1 .. i); g := proc (x) options operator, arrow; 0*x end proc; h := proc (x) options operator, arrow; 10000000*x end proc; plt[i] := plot(m(x), x = -r00 .. r00, y = -r11 .. r11, color = col1[i], thickness = 4, axes = boxed, title = 'Polynomials*x^(k+1)+2*x^k-1', titlefont = [COURIER, Bold, 14], labels = ["x", " m(x)"], labelfont = [COURIER, BOLDITALIC, 12], axesfont = [COURIER, 12], gridlines, tickmarks = [10, 10], legend = [i], legendstyle = [font = ["HELVETICA", 9], location = right]); pl1 := plot(g(x), x = -r00 .. r00, y = -r11 .. r11, color = black, thickness = 2, axes = boxed, title = 'Polynomials = x^(k+1)+2*x^k-1', titlefont = [COURIER, Bold, 14], labels = ["x", "m(x)"], labelfont = [COURIER, BOLDITALIC, 12], axesfont = [COURIER, 12], gridlines, tickmarks = [10, 10]); pl2 := plot(h(x), x = -r00 .. r00, y = -r11 .. r11, color = black, thickness = 2, axes = boxed, title = 'Polynomials = x^(k+1)+2*x^k-1', titlefont = [COURIER, Bold, 14], labels = ["x", "m(x)"], labelfont = [COURIER, BOLDITALIC, 12], axesfont = [COURIER, 12], gridlines, tickmarks = [10, 10]) end do; display({pl1, pl2, seq(plt[j], j = 2 .. n)})
```

1.6 City Grid Design



1.7 Ancient Monuments

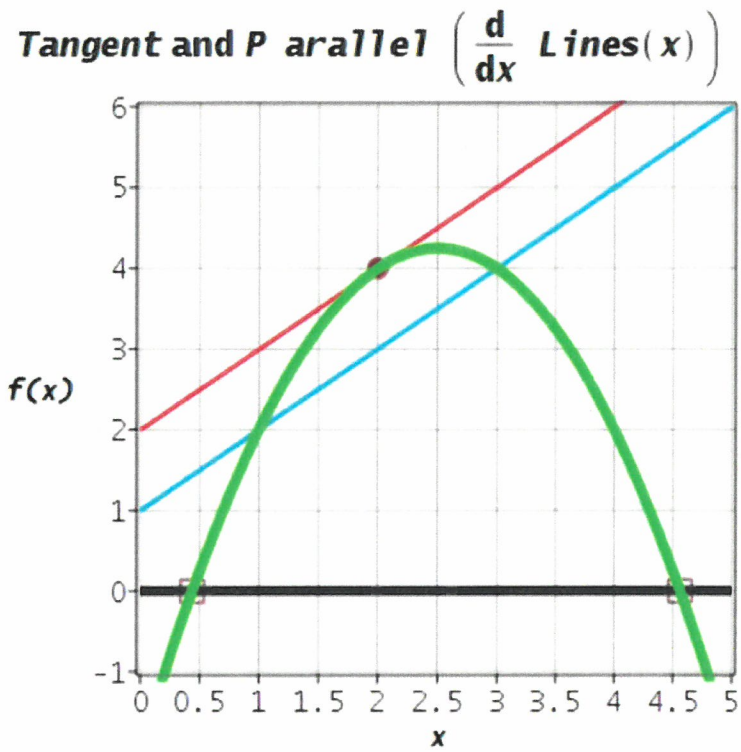
Ancient Monuments – Cities



restart; with(Student); with(plots); with(LinearAlgebra); with(RandomTools); n := 45; col1 := seq(COLOR(RGB, rand()/10^12, rand()/10^12, rand()/10^12), I = 1 .. n); lat1 := vector(45, Vector[column](45, [.258509265, .216261751, .240860488, .42927772, 0.61048595e-1, -.240858422, -.225801707, -.425376852, -.396607122, -.37067725, -.293687143, .265133166, -.233679239, -.42492514, -.30331112, -.533731764, -0.44926713e-1, .222267317, .218313309, 0.43072168e-1, -.241631982, -.255434957, -.289921942, -.301082779, -.308616469, -.559741263, -.572640075, .442351234, .167929381, -0.55265592e-1, 0.66997757e-1, .146921779, .245482199, .255309753, .326906347, .330393023, .376671487, -.350184702, -.427078035, -.361074684, -0.24683178e-1, .24718693, .254025434, .57429465, -.323089755])); long1 := vector(45, Vector[column](45, [.855816237, .299319453, .253195848, -.331610986, -.98419611, -.253198802, -.239949354, .370013005, .603668306, .693188859, .811231122, .363252345, .842606467, .122442689, .816974406, -.325571674, -.928162509, .309294748, .304588458, .193950308, 0.7855246e-2, -0.9306539e-2, -0.39258582e-1, -0.34316934e-1, -0.50881693e-1, -.308738709, -.317579431, -.521752781, -.260503018, .995312156, .993305742, .968074824, .905198443, .896744232, .812762244, .806788819, .70504736, 0.41004827e-1, .186525443, 0.6401283e-2, .999497375, .915662727, .903584836, -.444833703, -.144405029])); Radius1 := vector(45, Vector[column](45, [1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1])); pl1 := [seq(pointplot3d([Radius1[i], long1[i], lat1[i]], axes = boxed, symbol = solidcircle, symbolsize = 16, title =

'Ancient*Monuments-Cities', titlefont = [COURIER, Bold, 14], color = col1[i], coords = rectangular), I = 1 .. 45)]; display(pl1);

1.8 Parabola and Tangent Line at a Point



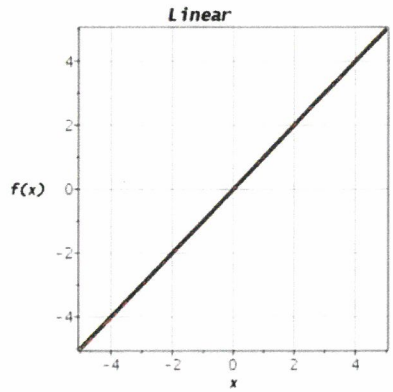
restart; with(plots); with(Student); with(RandomTools); f := proc (x) options operator, arrow; -x^2+5*x-2 end proc; g := proc (x) options operator, arrow; x+2 end proc; h := proc (x) options operator, arrow; x+1 end proc; k := proc (x) options operator, arrow; (1/10000000)*x end proc; eq1 := f(x) = 0; sol := solve(eq1, x); r1 := evalf(sol[1]); r2 := evalf(sol[2]); p0 := plot([[2, 4]], x = 0 .. 5, y = -1 .. 6, style = point, symbol = solidcircle, symbolsize = 20, axes = boxed, title = 'Tangent and P*arallel*Lines(x)', titlefont = [COURIER, Bold, 14], labels = ["x", "f(x)"], labelfont = [COURIER, BOLDITALIC, 12], axesfont = [COURIER, 12], gridlines, tickmarks = [10, 10]); p1 := plot([[r2, 0], [r1, 0]], x = 0 .. 5, y = -1 .. 6, style = point, symbol = box, symbolsize = 26, axes = boxed, title = 'Tangent and P*arallel*(diff(Lines(x), x))', titlefont = [COURIER, Bold, 14], labels = ["x", "f(x)"], labelfont = [COURIER, BOLDITALIC, 12], axesfont = [COURIER, 12], gridlines, tickmarks = [10, 10]); p2 := plot({f(x), k(x)}, x = 0 .. 5, y = -1 .. 6, color = [black, green], thickness = 5, axes = boxed, title = 'Tangent and P*arallel*(diff(Lines(x), x))', titlefont = [COURIER, Bold, 14], labels = ["x", "f(x)"], labelfont = [COURIER, BOLDITALIC, 12], axesfont = [COURIER, 12], gridlines, tickmarks = [10, 10]); p3 := plot(g(x), x = 0 .. 5, y = -1 .. 6, color = red, thickness = 2, axes = boxed, title = 'Tangent and P*arallel*(diff(Lines(x), x))', titlefont = [COURIER, Bold, 14], labels = ["x", "f(x)"], labelfont = [COURIER, BOLDITALIC, 12], axesfont = [COURIER, 12], gridlines, tickmarks = [10, 10]); p4 := plot(h(x), x = 0 .. 5, y = -1 .. 6, color = cyan, thickness = 2, axes = boxed, title = 'Tangent and P*arallel*(diff(Lines(x), x))', titlefont

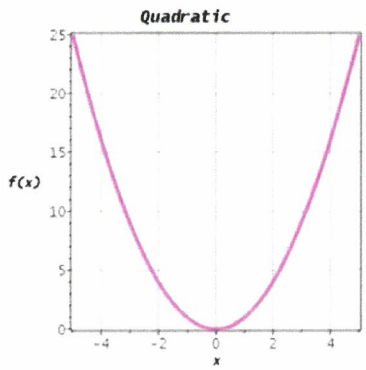
= [COURIER, Bold, 14], labels = ["x", "f(x)"], labelfont = [COURIER, BOLDITALIC, 12], axesfont = [COURIER, 12], gridlines, tickmarks = [10, 10]); display({p0, p1, p2, p3, p4})

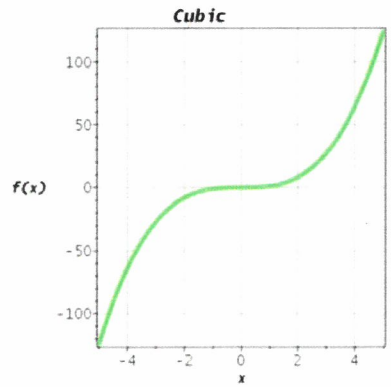
1.9 Functions, Differentials, Integrals, & Riemann Sums

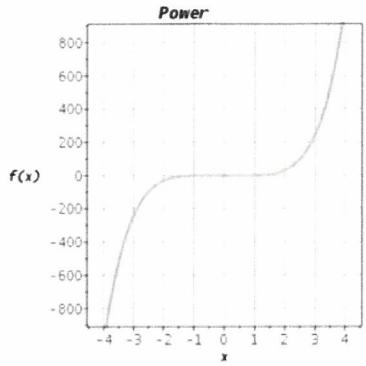
$$n := 5$$

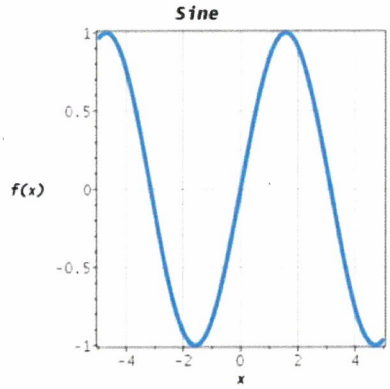
$$func1 := [x, x^2, x^3, x^5, \sin(x), \cos(x), \tan(x), e^x, \ln(x), \arcsin(x)]$$

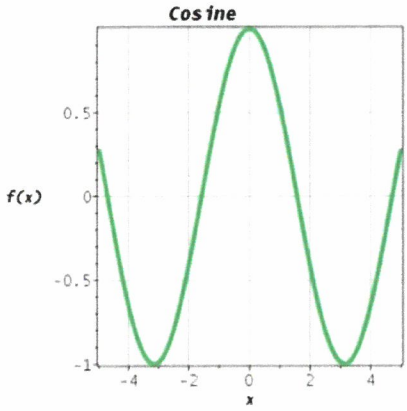


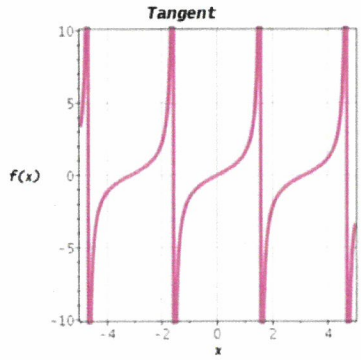


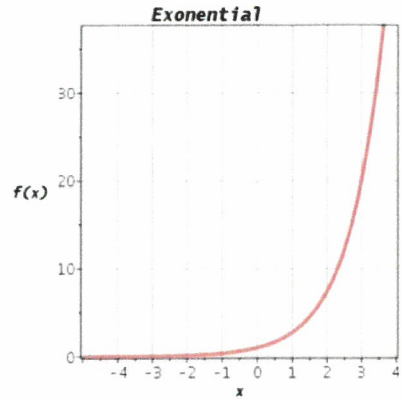


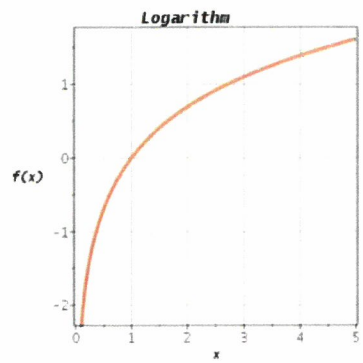

Power
f(x)
x

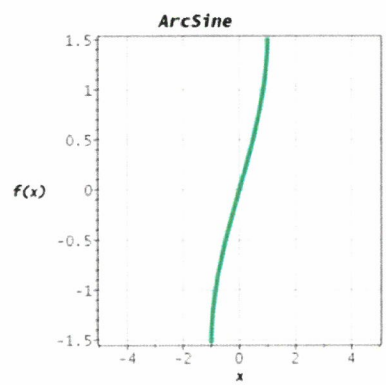

Sine
f(x)
x


Cosine
f(x)
x


Tangent
f(x)
x

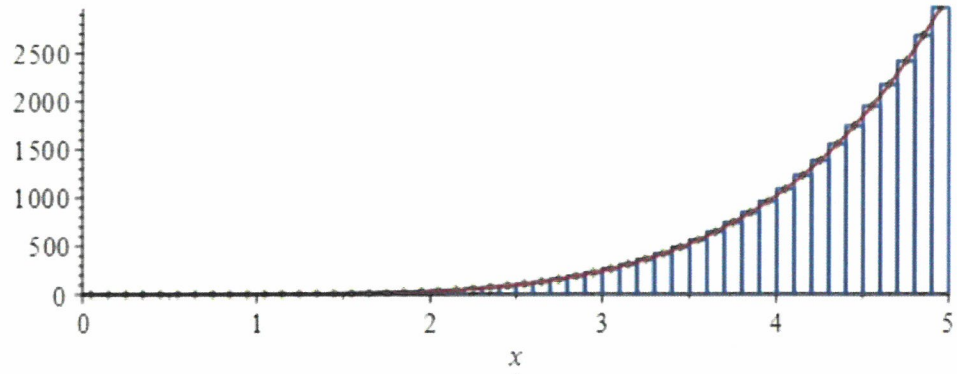




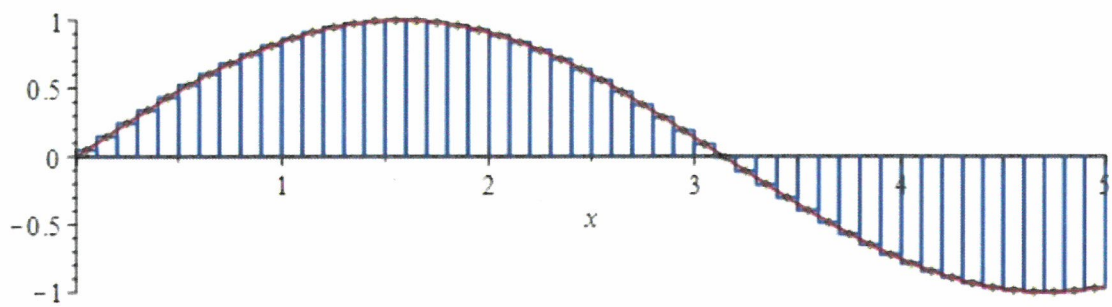


$$1,\ 2\,x,\ 3\,x^2,\ 5\,x^4,\ \cos(x),\ -\sin(x),\ 1+\tan(x)^2,\ e^x,\ \frac{1}{x},\ \frac{1}{\sqrt{-x^2+1}}$$

$$\frac{x^2}{2},\ \frac{x^3}{3},\ \frac{x^4}{4},\ \frac{x^6}{6},\ -\cos(x),\ \sin(x),\ -\ln(\cos(x)),\ e^x,\ x\ln(x)-x,\ x\arcsin(x)+\sqrt{-x^2+1}$$



A midpoint Riemann sum approximation of $\int_0^5 f(x)\,dx$, where $f(x) = x(x)^5$ and the partition is uniform. The approximate value of the integral is 2602.864766. Number of subintervals used: 50.



A midpoint Riemann sum approximation of $\int_0^5 f(x)\,dx$, where $f(x) = \sin(x)(x)$ and the partition is uniform. The approximate value of the integral is 0.7166363756. Number of subintervals used: 50.

$$f1 := x \mapsto \sin(x)$$

$$a := 0.$$

$$b0 := 1.570796325$$

$$b := 5.0$$

$$NT := 50$$

$$dh := 0.1000000000$$

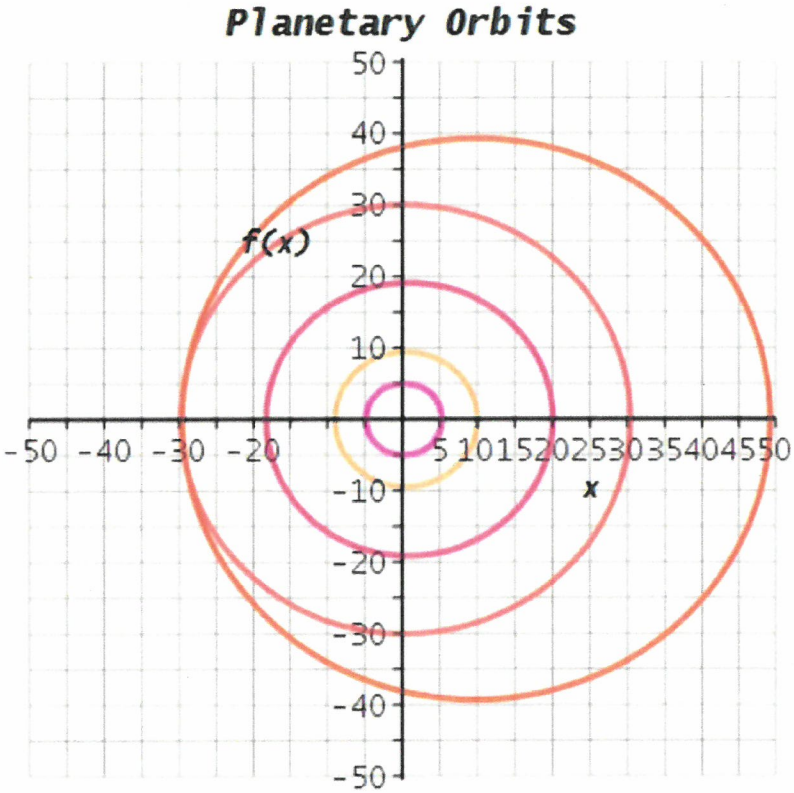
$$a1 := 0.$$

$$b1 := -0.9589242747$$

$$Intxtrap := 0.7157407668$$

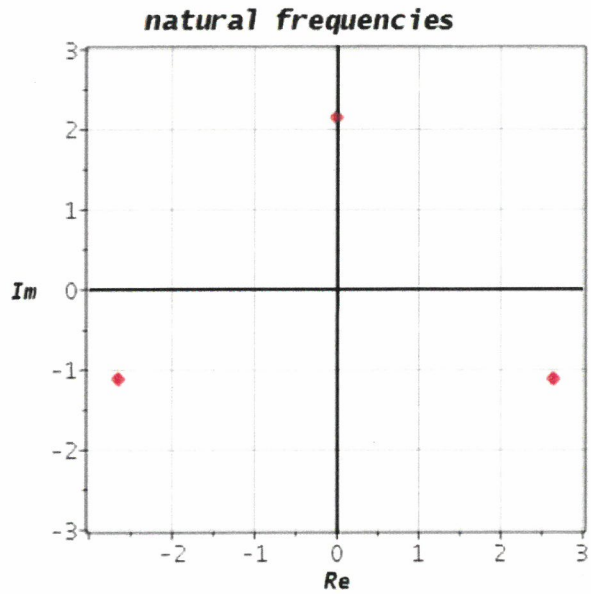
$$IntxRS := 0.7540977378$$

2.0 Planetary Orbits




```
restart; with(plots); with(Student); with(RandomTools); r1 := 1.; n1 := 9; col1 := seq(COLOR(RGB,
rand()/10^11, rand()/10^12, rand()/10^12), j = 1 .. n1); a := [.387, .723, 1., 1.52, 5.2, 9.54, 19.18, 30.07,
39.44]; b := [.264664, .719491, .991464, 1.472766, 5.175944, 9.511959, 19.15699, 30.065, 39.3158];
epsilon := [.206, 0.7e-2, 0.17e-1, 0.93e-1, 0.48e-1, 0.56e-1, 0.46e-1, 0.1e-1, .248]; c1 :=
seq(evalf(epsilon[i]*a[i]), I = 1 .. n1); x1 := 50; for I to n1 do pl[i] := implicitplot((x-
c1[i])^2/a[i]^2+y^2/b[i]^2-r1^2, x = -x1 .. x1, y = -x1 .. x1, color = col1[i], thickness = 3, axes = normal,
title = 'Planetary*Orbits', titlefont = [COURIER, Bold, 14], labels = ["x", "f(x)"], labelfont = [COURIER,
BOLDITALIC, 12], axesfont = [COURIER, 12], gridlines, tickmarks = [20, 20], gridrefine = 1) end do; g :=
proc (x) options operator, arrow; 0*x end proc; h := proc (x) options operator, arrow; 10000000*x end
proc; pl2 := plot(g(x), x = -x1 .. x1, y = -x1 .. x1, color = black, thickness = 2, axes = normal, title =
'Planetary*Orbits', titlefont = [COURIER, Bold, 14], labels = ["x", "f(x)"], labelfont = [COURIER,
BOLDITALIC, 12], axesfont = [COURIER, 12], gridlines, tickmarks = [20, 20]); pl3 := plot(h(x), x = -x1 .. x1, y
= -x1 .. x1, color = black, thickness = 1, axes = normal, title = 'Planetary*Orbits', titlefont = [COURIER,
Bold, 14], labels = ["x", "f(x)"], labelfont = [COURIER, BOLDITALIC, 12], axesfont = [COURIER, 12],
gridlines, tickmarks = [20, 20]); display({pl2, pl3, seq(pl[j], j = 1 .. n1)});
```

2.1 Bicycle Natural Frequencies




```
restart; with(Student); with(LinearAlgebra); with(DETools); with(plots); A0 := Matrix(2, 2, [[3., -0.5e-2], [-
0.2e-2, 0.5e-1]]); A0 := Matrix(2, 2, [[80.81722, 2.31941332208709], [2.31941332208709,
.29784188199686]]); g := 9.81; v := 3.5; omega1 := 10; A := v . Matrix(2, 2, [[0, 33.86641391492494], [-
.85035641456978, 1.68540397397560]]); K0 := Matrix(2, 2, [[-80.95, -2.59951685249872], [-
2.59951685249872, -.80329488458618]]); K2 := Matrix(2, 2, [[0, 76.59734589573222], [0,
2.65431523794604]]); B := evalf(K2*v^2+K0*g); q := Matrix(2, 1, [x(t), y(t)]); zer0 := Matrix(2, 1, [0, 0]);
zer1 := Matrix(2, 1, [exp(-t)*sin(omega1*t), 0]); Id1 := IdentityMatrix(2); ampl := Matrix(2, 2, [[a0, 0], [0,
b0]]); qs := Matrix(2, 1, [exp(a*t), exp(a*t)]); q2 := ampl . qs; q[1] := q2[1]; q[2] := q2[2]; de1 := A0 .
(diff(q, t, t))+A . (diff(q, t))+B . q; mult := Matrix(2, 2, [[exp(-a*t), 0], [0, exp(-a*t)]]); sys0 := evalf(mult .
de1); sys1 := simplify(sys0[1]); sys2 := simplify(sys0[2]); sys1f := collect(sys1, [a0, b0]); sys2f :=
collect(sys2, [a0, b0]); m11 := coeff(sys1f, a0); m21 := coeff(sys2f, a0); m12 := coeff(sys1f, b0); m22 :=
coeff(sys2f, b0); mat := Matrix(2, 2, [[m11, m12], [m21, m22]]); p := Determinant(%); poly := subs(a =
I*omega, p); natf := [solve(poly = 0, omega)]; natf[1]; for I to 4 do pl[i] := complexplot([natf[i]], -3 .. 3, -3
.. 3, style = point, symbol = soliddiamond, symbolsize = 16, color = red, gridlines, axes = boxed, title =
'natural*frequencies', titlefont = [COURIER, Bold, 14], labels = ["Re", "Im"], labelfont = [COURIER,
BOLDITALIC, 12], axesfont = [COURIER, 12]) end do; g := proc (x) options operator, arrow; 0*x end proc;
h := proc (x) options operator, arrow; 10000000*x end proc; plx := plot(g(x), x = -3 .. 3, y = -3 .. 3, style =
line, thickness = 2, color = black); ply := plot(h(x), x = -3 .. 3, y = -3 .. 3, style = line, thickness = 2, color =
black); plt := plots[display](seq(pl[i], I = 1 .. 4), plx, ply);
```

2.2 Linear Bicycle Dynamics

$$\underline{\underline{M}}\,\ddot{\underline{V}} + v\underline{\underline{A}}\dot{\underline{V}} + \left(g\underline{\underline{K}}_0 + v^2\underline{\underline{K}}_2\right)\underline{V} = \underline{f}$$

$$\underline{\underline{M}} = \begin{pmatrix} m_{11} & m_{12} \\ m_{21} & m_{22} \end{pmatrix}$$

$$\underline{\underline{A}} = \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix}$$

$$\underline{\underline{K}}_0 = \begin{pmatrix} k_{11} & k_{12} \\ k_{21} & k_{22} \end{pmatrix}$$

$$\underline{\underline{K}}_2 = \begin{pmatrix} K_{11} & K_{12} \\ K_{21} & K_{22} \end{pmatrix}$$

$$\underline{V} = \begin{pmatrix} \phi \\ \delta \end{pmatrix},\ \dot{\underline{V}} = \begin{pmatrix} \dot{\phi} \\ \dot{\delta} \end{pmatrix},\ \ddot{\underline{V}} = \begin{pmatrix} \ddot{\phi} \\ \ddot{\delta} \end{pmatrix},\ \underline{f} = \begin{pmatrix} f_1 \\ f_2 \end{pmatrix}$$

lean ϕ , steer δ

$$m_{11}\ddot{\phi} + m_{12}\ddot{\delta} + v\left(a_{11}\dot{\phi} + a_{12}\dot{\delta}\right) + g(k_{11}\phi + k_{12}\delta) + v^2(K_{11}\phi + K_{12}\delta) = f_1 \rightarrow e^{-\frac{t}{\tau}}\sin(\omega t + \theta)$$

$$m_{21}\ddot{\phi} + m_{22}\ddot{\delta} + v\left(a_{21}\dot{\phi} + a_{22}\dot{\delta}\right) + g(k_{21}\phi + k_{22}\delta) + v^2(K_{21}\phi + K_{22}\delta) = f_2 \rightarrow 0$$

$$\phi(0) = \phi_0,\ \delta(0) = \delta_0\ ,\ \dot{\phi}(0) = 0\ ,\ \dot{\delta}(0) = 0$$

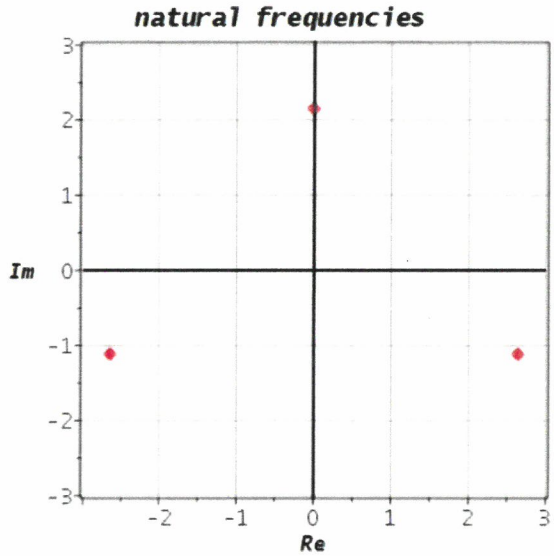
$$eq1 := \left[80.81722\frac{d^2}{dt^2}x(t) + 2.31941332208709\frac{d^2}{dt^2}y(t) + 33.8664139149249\frac{d}{dt}y(t) - 794.119500000000\,x(t) + 51.0960855727198\,y(t) - e^{-t}\sin(10\,t)\right]$$

$$eq2 := \left[2.31941332208709\frac{d^2}{dt^2}x(t) + 0.29784188199686\frac{d^2}{dt^2}y(t) - 0.850356414569780\frac{d}{dt}x(t) + 1.68540397397560\frac{d}{dt}y(t) - 25.5012603230124\,x(t) - 5.22600757984439\,y(t)\right]$$

x(0) := 5., y(0) := 1., D(x)(0) := 0, D(y)(0) := 0

Ref Meijaard, Jaap P., Jim M. Papadopoulos, Andy Ruina, and Arend L. Schwab. "Linearized dynamics equations for the balance and steer of a bicycle: a benchmark and review." In *Proceedings of the*

Royal Society of London A: Mathematical, Physical and Engineering Sciences, vol. 463, no. 2084, pp. 1955-1982. The Royal Society, 2007.

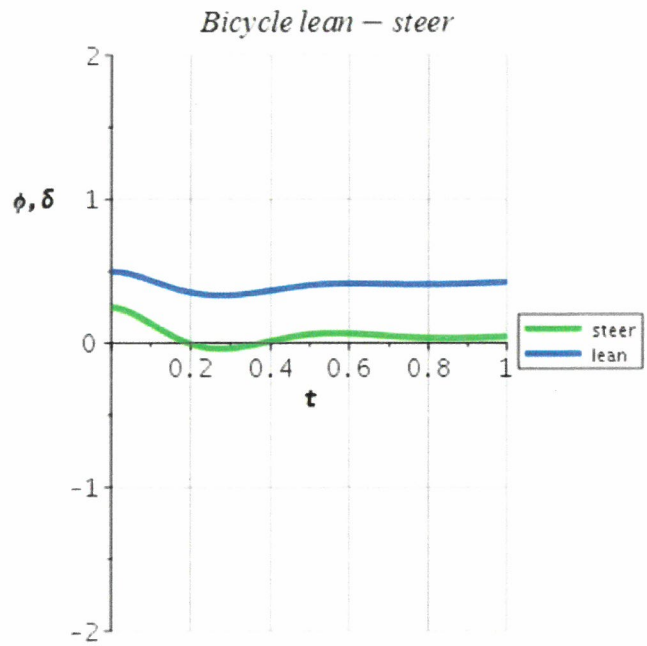


$$poly := 18.69107474\,\omega^4 - 208.7112305\,\mathrm{I}\,\omega^3 - 49.145052\,\omega^2 + 1055.051236\,\mathrm{I}\,\omega + 3714.79950$$

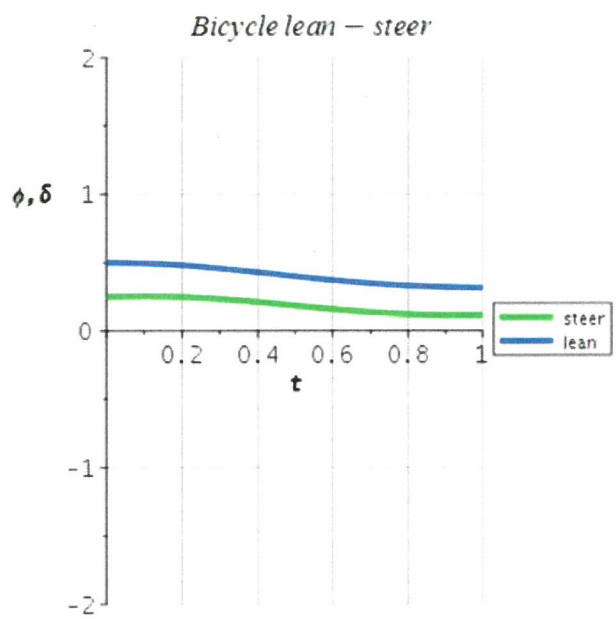
natf := [2.644144863 - 1.111251629 I, 2.149580430 I, 11.23928075 I, -2.644144863 - 1.111251629 I]

$$eq1 := \left[80.81722\,\frac{d^2}{dt^2}\,x(t) + 2.31941332208709\,\frac{d^2}{dt^2}\,y(t) + 338.664139149249\,\frac{d}{dt}\,y(t) - 794.119500000000\,x(t) + 7634.23332925021\,y(t) - 0.5\,e^{-t}\sin(1.5\,t)\right]$$

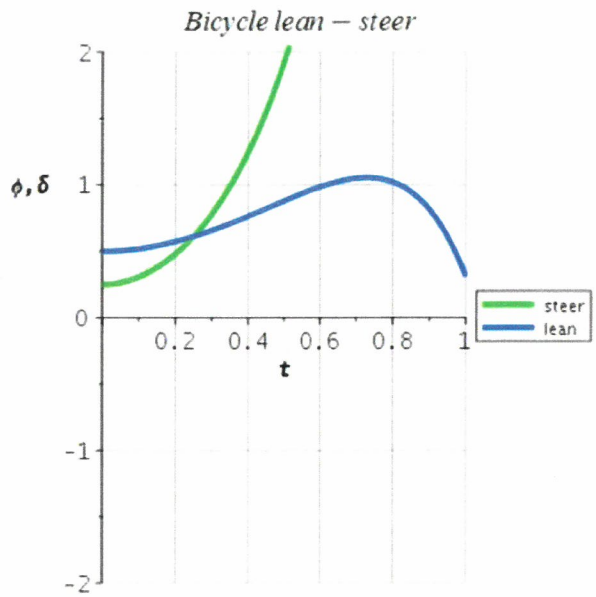
$$eq2 := \left[2.31941332208709\,\frac{d^2}{dt^2}\,x(t) + 0.29784188199686\,\frac{d^2}{dt^2}\,y(t) - 8.50356414569780\,\frac{d}{dt}\,x(t) + 16.8540397397560\,\frac{d}{dt}\,y(t) - 25.5012603230124\,x(t) + 257.551200976814\,y(t)\right]$$


Bicycle lean − steer
2
1
0
-1
-2
ϕ, δ
0.2
0.4
0.6
0.8
1
t
steer
lean

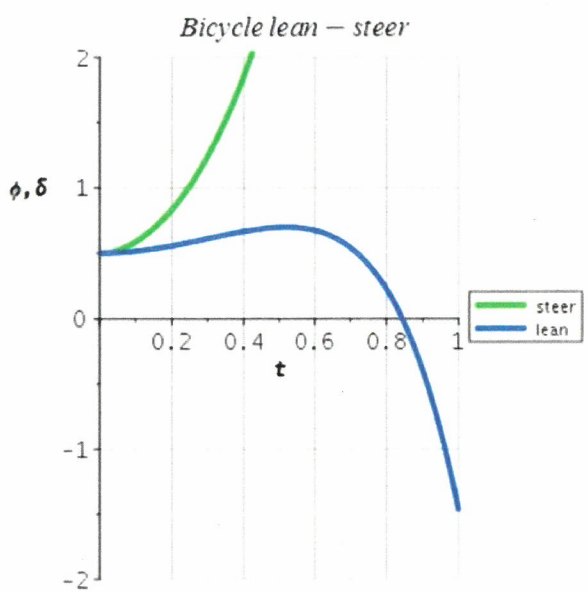
$v = 10.$


Bicycle lean − steer
2
1
0
-1
-2
ϕ, δ
0.2
0.4
0.6
0.8
1
t
steer
lean

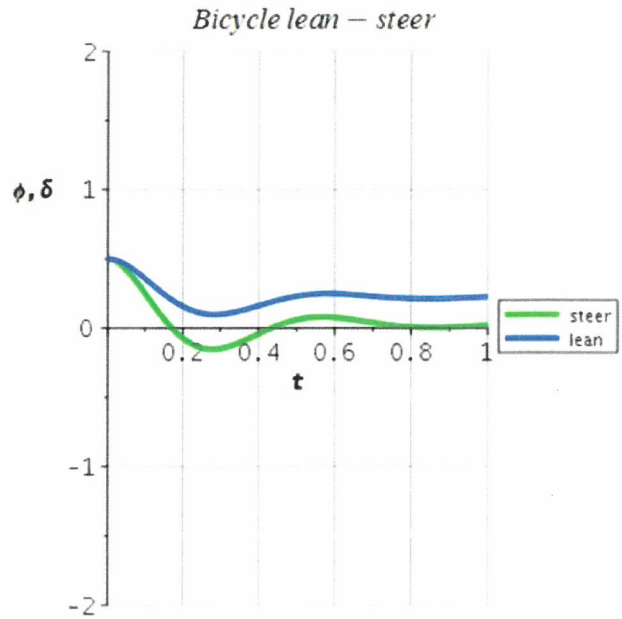
$v = 5..$



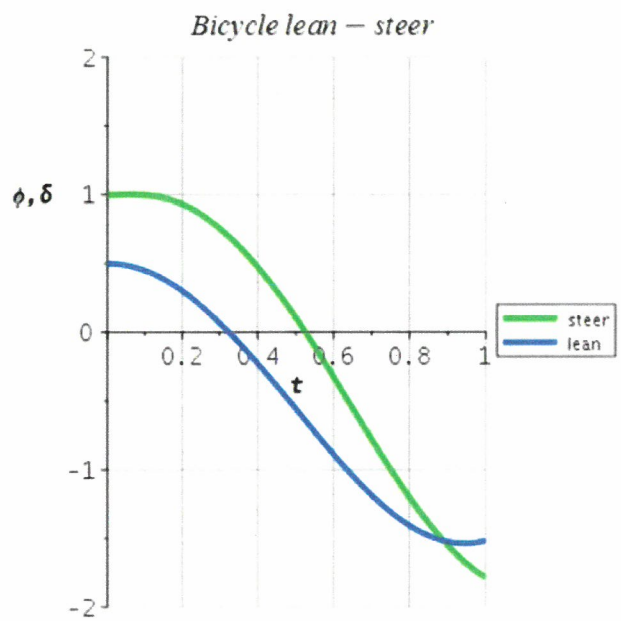
$v = 1.5.$



$v = 1.5., x(0) = 0.5 = y(0)$

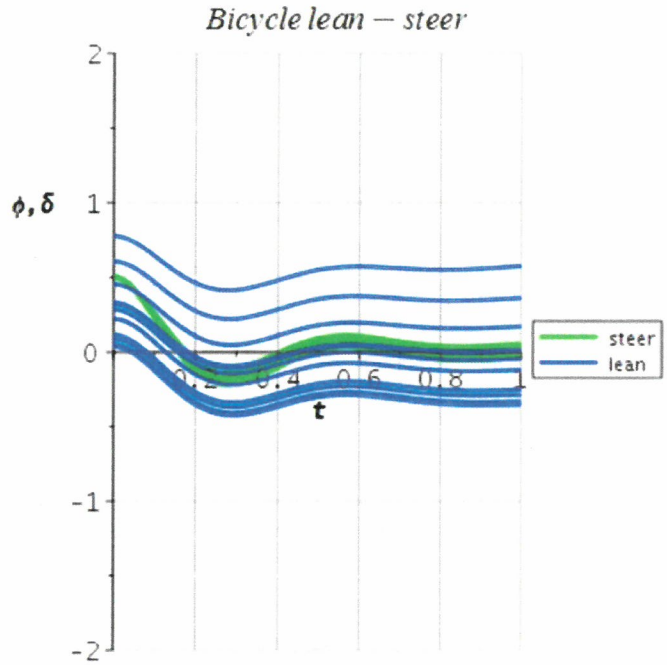


$v = 10.., x(0) = 0.5 = y(0), \omega = 1.5$



$v = 4.$

Random amplitude and phase



$$v = 10.., x(0) = 0.5 = y(0), \omega = 1.5$$

$$f_1 = \exp\left(\frac{-t}{\tau}\right)\left(1 - \frac{t}{\tau}\right) \mathrm{A(i)} \sin(\omega t + \theta(i)); \text{ sample i}$$

$$\mathrm{A(i)} \in [0.5,1.5],\ \theta(\mathrm{i}) \in [0.5,1.5]$$

2.3 Fractional Calculus Generalization

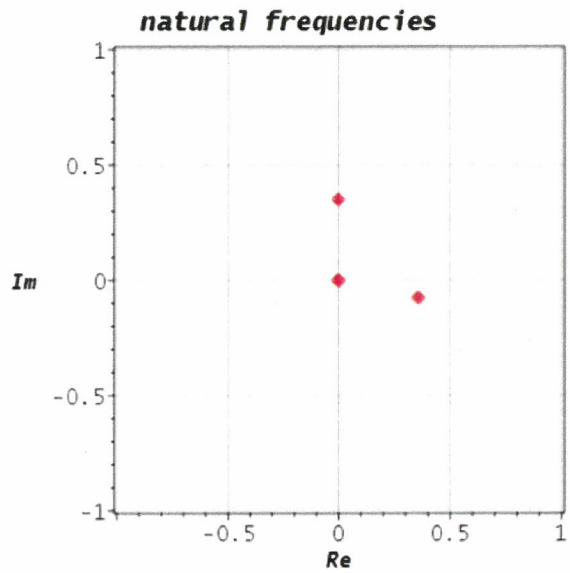
$$\underline{\underline{M}}\,\ddot{\underline{V}} + v\underline{\underline{A}}\,\frac{d^{0.5}}{dt^{0.5}}\underline{V} + \left(g\underline{\underline{K_0}} + v^2\underline{\underline{K_2}}\right)\underline{V} = \underline{f}$$

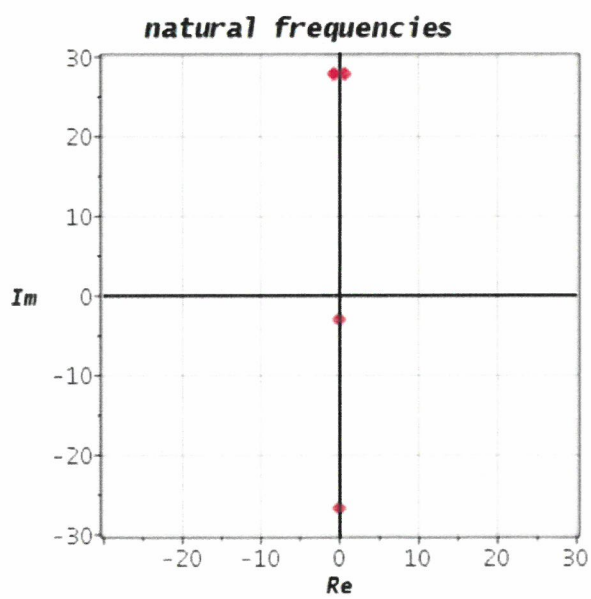
$$poly := 1.\omega^4 + 769.4844612\,\omega^2 - 5.898913909\,\omega^2\sqrt{\mathrm{I}\omega} + 3714.79950 + 1055.051236\sqrt{\mathrm{I}\omega} + 352.7818983\,\mathrm{I}\omega$$

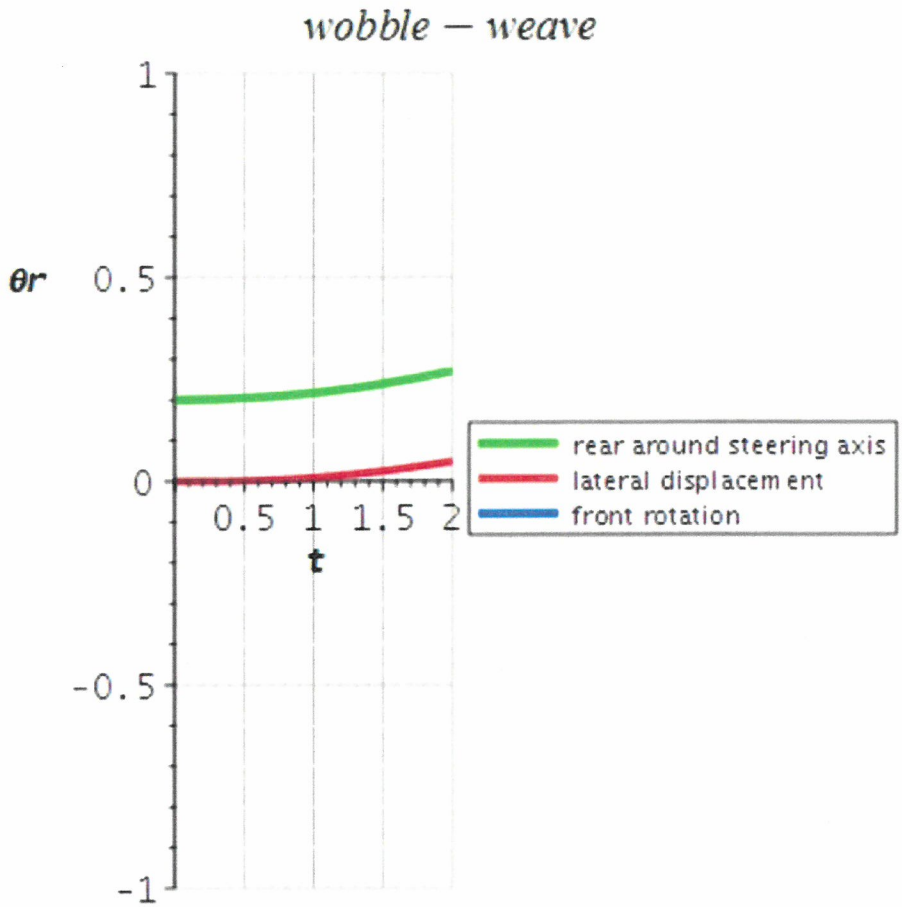
$$natff := -2.959457495\,\mathrm{I},\ -26.70080877\,\mathrm{I},\ 0.6854949836 + 27.88745367\,\mathrm{I},\ -0.6854949836 + 27.88745367\,\mathrm{I}$$

$$poly := -13554.35232\,\omega^6 + 73884.9139\,I\,\omega^5 + 15318.11212\,\omega^4 - 6418.56323\,I\,\omega^3 - 3344.379138\,\omega^2$$

$$natf := [0., 0., 0.3583029063 - 0.07537103514\,I, 0.3504834739\,I, 5.251268881\,I, -0.3583029063 - 0.07537103514\,I]$$







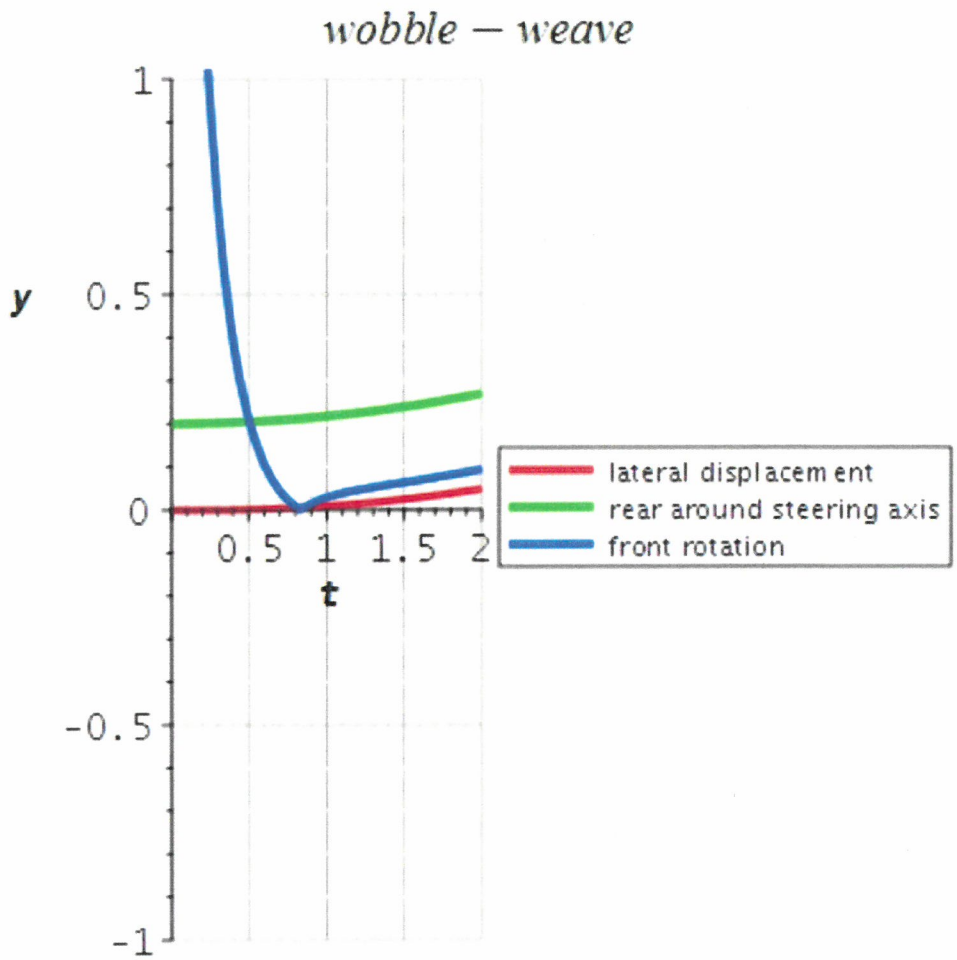
$$v = 30$$

$$\left[248.2 \frac{d^2}{dt^2} x(t) - 145.725 \frac{d^2}{dt^2} y(t) + 0.7368 \frac{d^2}{dt^2} z(t) + 0.899999999910000 \frac{d}{dt} x(t) - 0.726799999927320 \frac{d}{dt} y(t) - 0.0433066666623360 \frac{d}{dt} z(t) - 14.07790594 y(t) - 9.979275095 z(t)\right]$$

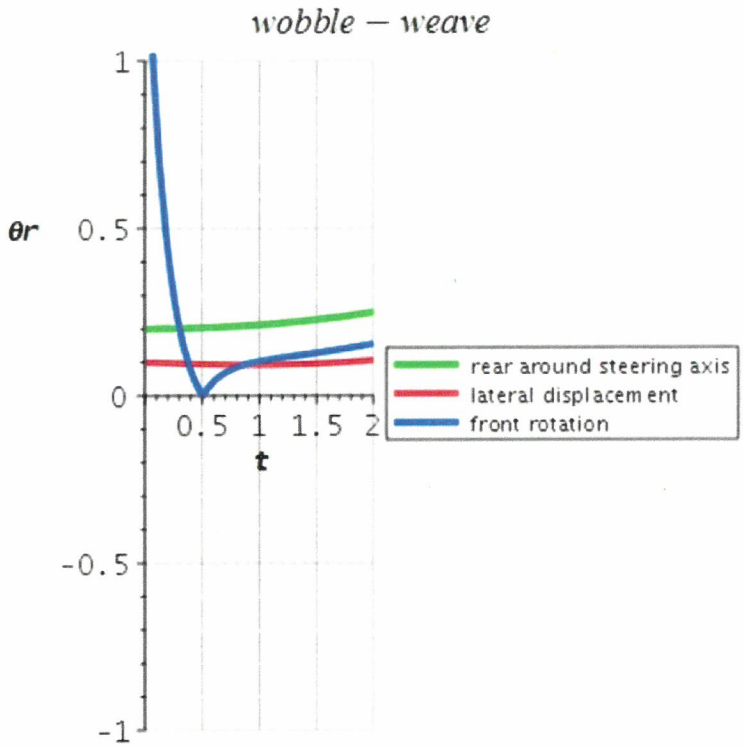
$$-145.725 \frac{d^2}{dt^2} x(t) + 21. \frac{d^2}{dt^2} y(t) - 0.726799999927320 \frac{d}{dt} x(t) - 6.79999999932000 \frac{d}{dt} y(t) + 6.80502357265283 \frac{d}{dt} z(t) + 19.42751019 y(t)$$

$$\left[0.7368 \frac{d^2}{dt^2} x(t) + 1.2176832 \frac{d^2}{dt^2} z(t) - 0.0433066666623360 \frac{d}{dt} x(t) - 6.79999999932000 \frac{d}{dt} y(t) + 6.80502357265283 \frac{d}{dt} z(t) + 1.157595911 z(t)\right]$$

$$x(0) := 0.1$$
$$y(0) := 0.2$$
$$z(0) := 0.$$
$$D(x)(0) := 0$$
$$D(y)(0) := 0$$
$$D(z)(0) := 0$$



$$v = 10$$



$$v = 15$$

x(0) := 0.1

y(0) := 0.2

z(0) := 0.1

D(x)(0) := 0

D(y)(0) := 0

D(z)(0) := 0

```
restart; with(Student); with(LinearAlgebra); with(DETools); with(plots); with(inttrans); A0 := Matrix(2, 2, [[80.81722, 2.31941332208709], [2.31941332208709, .29784188199686]]); g := 9.81; v := 4.; omega1 := 1.5; A := v . Matrix(2, 2, [[0, 33.86641391492494], [-.85035641456978, 1.68540397397560]]); K0 := Matrix(2, 2, [[-80.95, -2.59951685249872], [-2.59951685249872, -.80329488458618]]); K2 := Matrix(2, 2, [[0, 76.59734589573222], [0, 2.65431523794604]]); B := evalf(K2*v^2+K0*g); q := Matrix(2, 1, [x(t), y(t)]); zer0 := Matrix(2, 1, [0, 0]); ic1 := .5; ic2 := .5; zer1 := Matrix(2, 1, [ic1*exp(-t)*sin(omega1*t)*(Heaviside(t)-Heaviside(t-1)), 0]); Id1 := IdentityMatrix(2); diff(q[1], t, t); diff(q[1], t); de1 := A0 . (diff(q, t, t))+A . (diff(q, t))+B . q-Id1 . zer1; eq1 := de1[1]; eq2 := de1[2]; x(0) := ic1; y(0) := ic2; (D(x))(0) := 0; (D(y))(0) := 0; delp1 := laplace(eq1[1], t, s); delp1s := collect(delp1, [laplace(x(t), t, s), laplace(y(t), t, s)], recursive); delp2 := laplace(eq2[1], t, s); delp2s := collect(delp2, [laplace(x(t), t, s),
```

```
laplace(y(t), t, s)], recursive); sol1 := solve({delp1s, delp2s}, {laplace(x(t), t, s), laplace(y(t), t, s)}); xt :=
invlaplace(rhs(sol1[1]), s, t); yt := invlaplace(rhs(sol1[2]), s, t); tm := 1; y1 := 2; pl1 := plot(evalf(xt), t = 0 ..
tm, y = -y1 .. y1, axes = normal, gridlines, thickness = 3, color = blue, style = line, title =
'Bicycle*Pulsed*lean*sinusoid-steer', titlefont = [COURIER, Bold, 14], labels = ["t", "&phi;,&delta;"],
labelfont = [COURIER, BOLDITALIC, 12], axesfont = [COURIER, 12], legend = ["lean"], titlefont = ["ARIAL",
15], legendstyle = [font = ["HELVETICA", 9], location = right]); pl2 := plot(evalf(yt), t = 0 .. tm, y = -y1 .. y1,
axes = normal, gridlines, thickness = 3, color = green, style = line, title = 'Bicycle*lean-steer', titlefont =
[COURIER, Bold, 14], labels = ["t", "&phi;,&delta;"], labelfont = [COURIER, BOLDITALIC, 12], legend =
["steer"], titlefont = ["ARIAL", 15], legendstyle = [font = ["HELVETICA", 9], location = right]);
plots[display]({pl1, pl2})
```

2.3 Simple Tire Model Dimensionless Parameters

```
restart; with(Student); with(combinat, randcomb); with(linalg); with(Student[LinearAlgebra]);
with(plots); m := 6; n1 := 3; dd := m-n1; N := factorial(m)/(factorial(m-n1)*factorial(n1)); m0 := [delta, w,
f, F, mu, R]; for j to N do mc[j] := randcomb(m0, n1) end do; M := [seq(mc[k], k = 1 .. N)]; for k to N do
m0s[k] := convert(m0, set); Mc[k] := convert(M[k], set); A0[k] := m0s[k] minus Mc[k]; Cm[k] := [A0[k],
Mc[k]]; V1[k] := Vector(dd, [seq(A0[k][j], j = 1 .. dd)]); V2[k] := Vector(n1, [seq(Mc[k][j], j = 1 .. n1)]);
V1tr[k] := Transpose(V1[k]); V2t[k] := Transpose(V2[k]) end do; V1tr[1]; V1tr[1](1, 1); V1tr[1](1, 2);
V2t[1]; V2t[1](1, 1); V2t[1](1, 2); V2t[1](1, 3); Am2[1] := [seq(A0[1][i], I = 1 .. dd)]; delta := Vector(3, [0, 1,
0]); w := Vector(3, [0, 1, 0]); f := Vector(3, [1, 1, -2]); F := Vector(3, [1, 1, -2]); mu := Vector(3, [1, 0, 0]); R
:= Vector(3, [0, 0, -1]); for k to N do AM1[k] := Matrix(n1, dd, [seq(V1tr[k][i], I = 1 .. dd)]); Bm1[k] :=
Matrix(n1, n1, [seq(V2t[k][i], I = 1 .. n1)]); det1[k] := Determinant(Bm1[k]) end do; for n to N do if det1[n]
= 0 then next end if; Bm1i[n] := inverse(Bm1[n]); I01[n] := -1.*Matrix(n1, n1, shape = identity); A1[n] :=
I01[n] . AM1[n]; Dim[n] := Bm1i[n] . A1[n] end do; for n to N do if det1[n] = 0 then next end if; dimpar[n]
:= seq(V2t[n](1, 1)^convert(Dim[n](1, i), rational)*V2t[n](1, 2)^convert(Dim[n](2, i), rational)*V2t[n](1,
3)^convert(Dim[n](3, i), rational)*V1tr[n](1, i), I = 1 .. n1) end do; RowDimension(V1tr[1]);
ColumnDimension(V1tr[1]); RowDimension(V2t[1]); ColumnDimension(V2t[1]); M[1]; V1[1]; V1tr[1];
V2[1]; V2t[1]; Dim[1]; Dim[1][1]; Dim[1][2]; Dim[1][3];
```

$$dimpar_1 := \frac{F}{R^2\,\mu\,w}, \frac{\delta}{w}, \frac{f}{R^2\,\mu\,w}$$

$$dimpar_3 := \frac{R^2\,\mu\,\delta}{F}, \frac{f}{F}, \frac{R^2\,\mu\,w}{F}$$

$$dimpar_4 := \frac{F}{f}, \frac{\sqrt{\delta}\,\sqrt{\mu}\,R}{\sqrt{f}}, \frac{w}{\delta}$$

$$dimpar_7 := \frac{R^2\,\mu\,\delta}{F}, \frac{f}{F}, \frac{R^2\,\mu\,w}{F}$$

$$dimpar_8 := \frac{F}{f}, \frac{\delta}{w}, \frac{R^2\,w\,\mu}{f}$$

$$dimpar_9 := \frac{\sqrt{\delta}\,\sqrt{\mu}\,R}{\sqrt{F}}, \frac{f}{F}, \frac{w}{\delta}$$

$$dimpar_{10} := \frac{F}{f}, \frac{R^2 \delta \mu}{f}, \frac{w}{\delta}$$

$$dimpar_{11} := \frac{R^2 \mu \delta}{F}, \frac{f}{F}, \frac{R^2 \mu w}{F}$$

$$dimpar_{12} := \frac{F}{f}, \frac{R^2 \delta \mu}{f}, \frac{R^2 w \mu}{f}$$

$$dimpar_{15} := \frac{F}{f}, \frac{\sqrt{\delta}\sqrt{\mu}\, R}{\sqrt{f}}, \frac{w}{\delta}$$

$$dimpar_{16} := \frac{F}{f}, \frac{\sqrt{\mu}\sqrt{w}\, R}{\sqrt{f}}, \frac{\delta}{w}$$

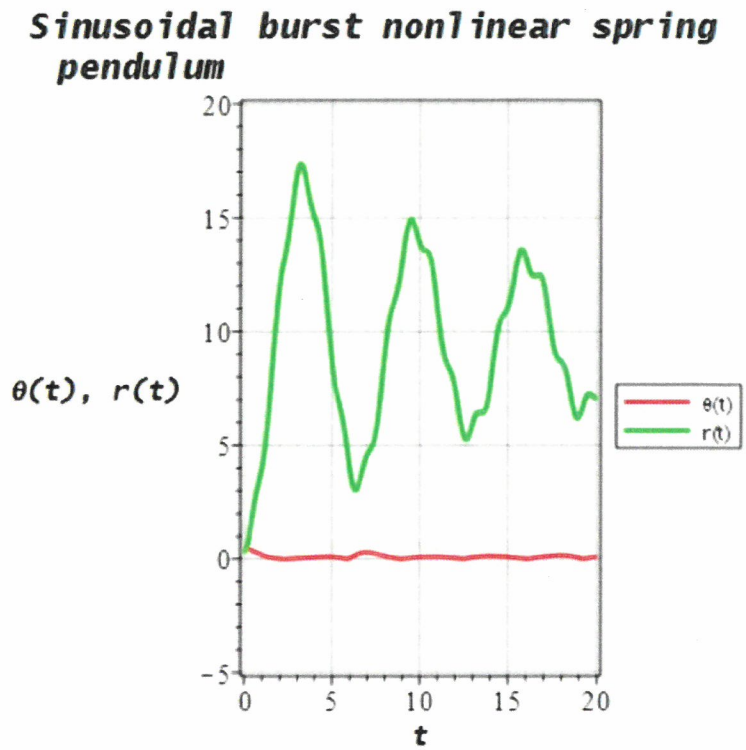
$$dimpar_{18} := \frac{\sqrt{\delta}\sqrt{\mu}\, R}{\sqrt{F}}, \frac{f}{F}, \frac{w}{\delta}$$

2.4 Nonlinear Spring-Pendulum Model

```
restart; with(DEtools); with(Student); with(plots); -linear*vibration+non; l1 := 2.; k1 := 10.; m := 10.;
`&omega;y` := 5.; a0 := .2; ys := a0*sin(`&omega;y`*t); a0 := .5; b0 := .5; omega1 := 1.5; `d&theta;` := .2; g
:= 9.81; dr := 1.; tau := 100.; DEqp1 := l1*(diff(theta(t), [t$2]))+r1(t)*(diff(theta(t), [t$2]))+2*(diff(theta(t),
t))*(diff(r1(t), t))-ys*sin(theta(t))+g*sin(theta(t))+`d&theta;`*(diff(theta(t), t))/m = exp(-t/tau)*(1-
t/tau)*a0*sin(omega1*t)*Heaviside(t); DEqp2 := diff(r1(t), [t$2])+dr*(diff(r1(t), t))/m+k1*r1(t)/m-
g*cos(theta(t))-a0*`&omega;y`^2*cos(`&omega;y`*t)*cos(theta(t))+l1*(diff(theta(t),
t))^2+r1(t)*(diff(theta(t), t))^2 = 0; init := eval(theta(t), t = 0) = a0, (D(theta))(0) = 0, eval(r1(t), t = 0) = b0,
(D(r1))(0) = 0; sol3 := dsolve({DEqp1, DEqp2, init}, {r1(t), theta(t)}, numeric, output = listprocedure,
abserr = 0.1e-2); T10 := 0.; T2 := 20.; y1 := 5.; y2 := 20; pl01 := odeplot(sol3, [t, abs(theta(t))], t = T10 ..
T2, numpoints = 500, labels = ["t", "&theta;(t), r(t)"], thickness = 2, title = ` Sinusoidal burst nonlinear
spring pendulum`, titlefont = [COURIER, Bold, 14], labelfont = [COURIER, BOLDITALIC, 12], color = red,
axes = boxed, view = [T10 .. T2, -y1 .. y2], legend = ["&theta;(t)"], legendstyle = [font = ["HELVETICA", 9],
location = right], gridlines); pl02 := odeplot(sol3, [t, abs(r1(t))], t = T10 .. T2, numpoints = 500, labels =
["t", "&theta;(t), r(t)"], thickness = 2, title = ` Sinusoidal burst nonlinear spring pendulum`, titlefont =
[COURIER, Bold, 14], labelfont = [COURIER, BOLDITALIC, 12], color = green, axes = boxed, view = [T10 ..
T2, -y1 .. y2], legend = ["r(t)"], legendstyle = [font = ["HELVETICA", 9], location = right], gridlines);
plots[display]({pl01, pl02});
```

$$DEqp1 := 2.\frac{d^2}{dt^2}\theta(t) + r1(t)\left(\frac{d^2}{dt^2}\theta(t)\right) + 2\left(\frac{d}{dt}\theta(t)\right)\left(\frac{d}{dt}r1(t)\right)$$
$$- 0.2\sin(5.t)\sin(\theta(t)) + 9.81\sin(\theta(t)) + 0.1000000000\frac{d}{dt}\theta(t)$$
$$= 0.5\,e^{-0.01000000000\,t}(1 - 0.01000000000\,t)\sin(1.5\,t)\,\mathrm{Heaviside}(t)$$

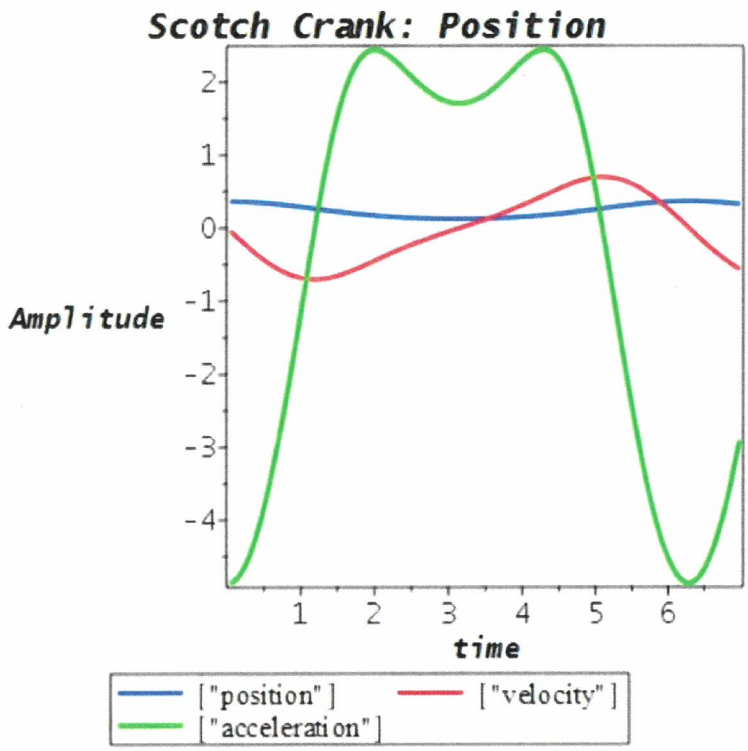
$$DEqp2 := \frac{d^2}{dt^2}\, r1(t) + 0.1000000000 \frac{d}{dt}\, r1(t) + 1.000000000\, r1(t) - 9.81 \cos(\theta(t)) - 12.5 \cos(5.\, t) \cos(\theta(t)) + 2. \left(\frac{d}{dt}\, \theta(t)\right)^2 + r1(t) \left(\frac{d}{dt}\, \theta(t)\right)^2 = 0$$



2.5 Scotch Crank

restart; with(RandomTools); THDrpm := 50; r = .12; c = .25; THD := evalf(2*THDrpm*Pi*(1/60)); tf := evalf(2*Pi/THD); t := seq(evalf(2*tf*i*(1/360)), I = 1 .. 360); TH := evalf(THD*t); d2s := seq(subs(c = .25, r = .12, c^2-r^2*sin(evalf(TH[i]))^2), I = 1 .. 200); x := seq(subs(r = .12, r*cos(TH[i])+sqrt(d2s[i])), I = 1 .. 200); xd := seq(subs(r = .12, evalf(-r*THD*sin(TH[i])-r^2*THD*sin(2*TH[i])/(2*sqrt(d2s[i])))), I = 1 .. 200); xdd := seq(subs(r = .12, evalf(-r*THD^2*cos(TH[i])-(4*r^2*THD^2*cos(2*TH[i])*d2s[i]+(r^2*THD*sin(2*TH[i]))^2)/(4*d2s[i]^1.5))), I = 1 .. 200); with(plots); pointplot({seq([TH[i], x[i]], I = 1 .. 200)}, axes = boxed, color = blue, style = line, thickness = 2, title = `Scotch Crank: Position`, titlefont = [COURIER, Bold, 14], labels = ["time", "Amplitude"], labelfont = [COURIER, BOLDITALIC, 12], axesfont = [COURIER, 12]); pointplot({seq([TH[i], xd[i]], I = 1 .. 200)}, axes = boxed, color = red, style = line, thickness = 2, title = `Scotch Crank: Velocity`, titlefont = [COURIER, Bold, 14], labels = ["time", "Amplitude"], labelfont = [COURIER, BOLDITALIC, 12], axesfont = [COURIER, 12]); pointplot({seq([TH[i], xdd[i]], I = 1 .. 200)}, axes = boxed, color = green, style = line, thickness = 2, title = `Scotch Crank: Acceleration`, titlefont = [COURIER, Bold, 14], labels = ["time", "Amplitude"], labelfont = [COURIER, BOLDITALIC, 12], axesfont = [COURIER, 12]); p1 := pointplot({seq([TH[i], x[i]], I = 1 .. 200)}, axes = boxed, color = blue, style = line, thickness = 2, title = `Scotch Crank: Position`, titlefont =

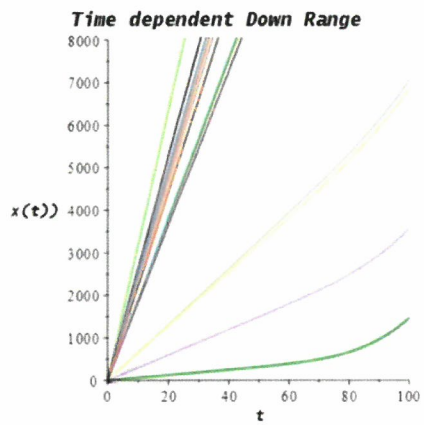
[COURIER, Bold, 14], labels = ["time", "Amplitude"], labelfont = [COURIER, BOLDITALIC, 12], axesfont = [COURIER, 12], legend = ["position"]); p2 := pointplot({seq([TH[i], xd[i]], I = 1 .. 200)}, axes = boxed, color = red, style = line, thickness = 2, title = `Scotch Crank: Velocity`, titlefont = [COURIER, Bold, 14], labels = ["time", "Amplitude"], labelfont = [COURIER, BOLDITALIC, 12], axesfont = [COURIER, 12], legend = ["velocity"]); p3 := pointplot({seq([TH[i], xdd[i]], I = 1 .. 200)}, axes = boxed, color = green, style = line, thickness = 2, title = `Scotch Crank: Acceleration`, titlefont = [COURIER, Bold, 14], labels = ["time", "Amplitude"], labelfont = [COURIER, BOLDITALIC, 12], axesfont = [COURIER, 12], legend = ["acceleration"]); plots[display]([p1, p2, p3]);

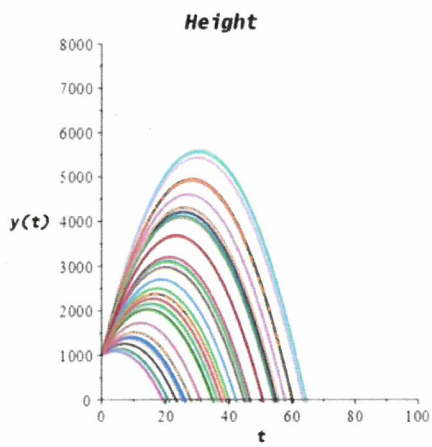


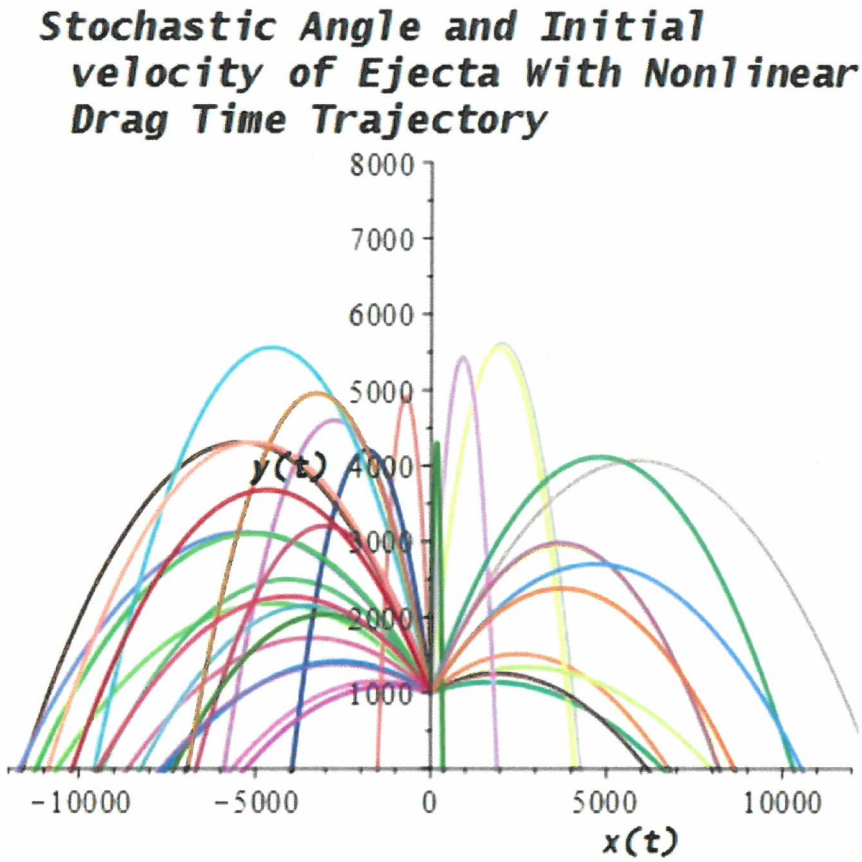
2.6 Volcanic Tephra

restart; with(DEtools); with(Student); with(plots); with(Statistics); with(RandomTools); R := 8.3143; T0 := 290; `ρd` := 1.6; rho0 := 1.225; r0 := 0.1e-2; phi1 := evalf((1/180)*(39*3.14159)); a1 := 0.1e-1; b1 := 1.; `gφ` := evalf(9.7803267714*(1+0.193185138639e-2*sin(phi1^2))/sqrt(1-0.6943779013e-2*sin(phi1^2))); g00 := evalf(`gφ`/(R*T0)); Cd := .4; Ch := .35; m0 := evalf((4/3*3.14159)*r0^3*`ρd`); A := evalf(3.14159*r0^2); n := 35; col1 := seq(COLOR(RGB, rand()/10^12, rand()/10^12, rand()/10^12), j = 1 .. n); theta0 := evalf(3.14159*(5*2)*(1/360)); theta11 := evalf(3.14159*(175*2)*(1/360)); `θr` := Generate(list(float(range = theta0 .. theta11, method = uniform), n)); v0 := Generate(list(float(range = 250 .. 350, method = uniform), n)); RE := 6378.137*10^3; g := 9.810; for I to n do de1 := diff(v(t), t)+.5*Cd*(1+a1*sqrt(v(t)^2+h(t)^2)^.637/Cd-b1/(1+1/sqrt(v(t)^2+h(t)^2)^1.09))*rho0*exp(-g00*h(t))*A*v(t)*sqrt(v(t)^2+h(t)^2)+.5*Ch*rho0*exp(-g00*h(t))*A*h(t)*sqrt(v(t)^2+h(t)^2) = 0; de2 := diff(h(t),

```
t)+`g&phi;`*RE/(RE+h(t))+.5*Cd*(1+a1*sqrt(v(t)^2+h(t)^2)^.637/Cd-
b1/(1+1/sqrt(v(t)^2+h(t)^2)^1.09))*rho0*exp(-g00*h(t))*A*h(t)*sqrt(v(t)^2+h(t)^2)-.5*Ch*rho0*exp(-
g00*h(t))*A*v(t)*sqrt(v(t)^2+h(t)^2) = 0; de3 := diff(x(t), t)-v(t) = 0; de4 := diff(y(t), t)-h(t) = 0; sol :=
dsolve({de1, de2, de3, de4, h(0) = v0[i]*sin(`&theta;r`[i]), v(0) = v0[i]*cos(`&theta;r`[i]), x(0) = 0, y(0) =
1000}, {h(t), v(t), x(t), y(t)}, numeric, output = listprocedure, abserr = 0.1e-2); T1 := 0.; T2 := 100; pl1[i] :=
odeplot(sol, [t, x(t)], t = T1 .. T2, numpoints = 1000, labels = ["t", "x(t))"], thickness = 2, title = ` Time
dependent Down Range `, titlefont = [COURIER, Bold, 14], titlefont = [COURIER, Bold, 14], labelfont =
[COURIER, BOLDITALIC, 12], color = col1[i], axes = normal, view = [T1 .. T2, 0 .. 8000]); pl2[i] :=
odeplot(sol, [[t, y(t)]], t = T1 .. T2, numpoints = 500, labels = ["t", " y(t)"], thickness = 2, title = ` Height`,
titlefont = [COURIER, Bold, 14], titlefont = [COURIER, Bold, 14], labelfont = [COURIER, BOLDITALIC, 12],
color = col1[i], axes = normal, view = [T1 .. T2, 0 .. 8000]); pl4[i] := odeplot(sol, [x(t), y(t)], t = T1 .. T2,
numpoints = 1000, labels = ["x(t)", "y(t)"], thickness = 2, title = ` Stochastic Angle and Initial velocity of
Ejecta With Nonlinear Drag Time Trajectory`, titlefont = [COURIER, Bold, 14], titlefont = [COURIER, Bold,
14], labelfont = [COURIER, BOLDITALIC, 12], color = col1[i], axes = normal, view = [-12000 .. 12000, 0 ..
8000]) end do; plots[display](seq(pl1[i], I = 1 .. n)); plots[display](seq(pl2[i], I = 1 .. n));
plots[display](seq(pl4[i], I = 1 .. n));
```

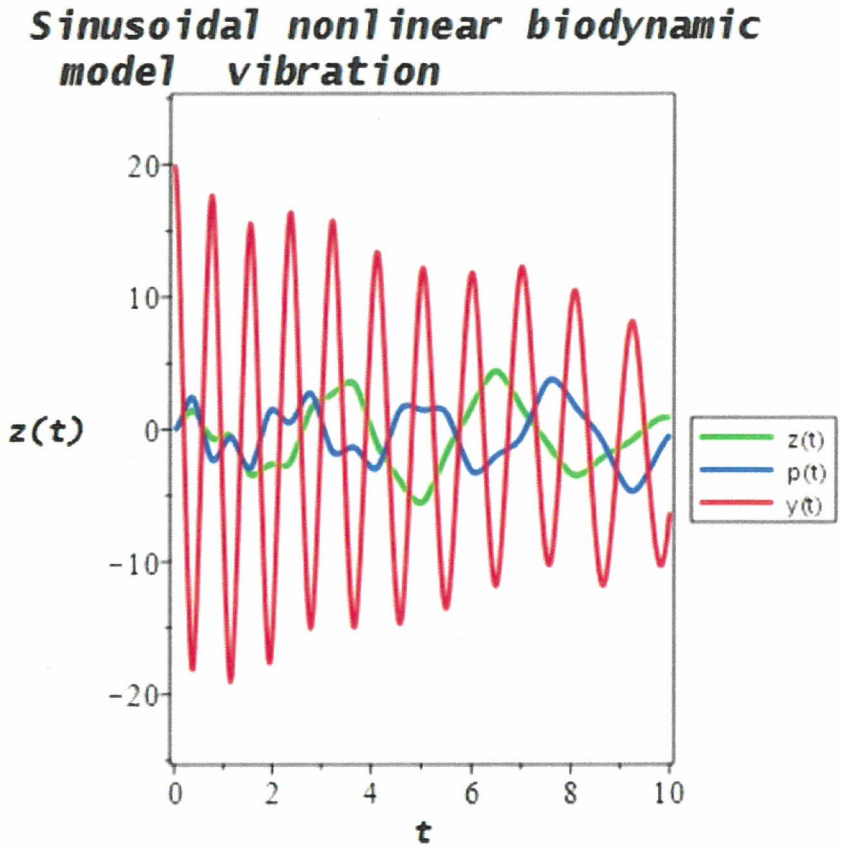






restart; with(plots); with(Student); with(RandomTools); r1 := 1.; n1 := 9; col1 := seq(COLOR(RGB, rand()/10^11, rand()/10^12, rand()/10^12), j = 1 .. n1); a := [.387, .723, 1., 1.52, 5.2, 9.54, 19.18, 30.07, 39.44]; b := [.264664, .719491, .991464, 1.472766, 5.175944, 9.511959, 19.15699, 30.065, 39.3158]; epsilon := [.206, 0.7e-2, 0.17e-1, 0.93e-1, 0.48e-1, 0.56e-1, 0.46e-1, 0.1e-1, .248]; c1 := seq(evalf(epsilon[i]*a[i]), I = 1 .. n1); x1 := 50; for I to n1 do pl[i] := implicitplot((x-c1[i])^2/a[i]^2+y^2/b[i]^2-r1^2, x = -x1 .. x1, y = -x1 .. x1, color = col1[i], thickness = 3, axes = normal, title = 'Planetary*Orbits', titlefont = [COURIER, Bold, 14], labels = ["x", "f(x)"], labelfont = [COURIER, BOLDITALIC, 12], axesfont = [COURIER, 12], gridlines, tickmarks = [20, 20], gridrefine = 1) end do; g := proc (x) options operator, arrow; 0*x end proc; h := proc (x) options operator, arrow; 10000000*x end proc; pl2 := plot(g(x), x = -x1 .. x1, y = -x1 .. x1, color = black, thickness = 2, axes = normal, title = 'Planetary*Orbits', titlefont = [COURIER, Bold, 14], labels = ["x", "f(x)"], labelfont = [COURIER, BOLDITALIC, 12], axesfont = [COURIER, 12], gridlines, tickmarks = [20, 20]); pl3 := plot(h(x), x = -x1 .. x1, y = -x1 .. x1, color = black, thickness = 1, axes = normal, title = 'Planetary*Orbits', titlefont = [COURIER, Bold, 14], labels = ["x", "f(x)"], labelfont = [COURIER, BOLDITALIC, 12], axesfont = [COURIER, 12], gridlines, tickmarks = [20, 20]); display({pl2, pl3, seq(pl[j], j = 1 .. n1)});

2.8 Biodynamic Model



$$DEq0 := 10\,\frac{d^2}{dt^2}\,y(t) + 2\,\zeta 1\,\omega\left(\frac{d}{dt}\,y(t)\right) + \omega^2\,y(t) + (y(t) - z(t))^2 + (y(t) - z(t))^3 + z(t)\,(y(t) - p(t))^2 + (y(t) - p(t))^3 = e^{-0.01000000000\,t}\,(1 - 0.01000000000\,t)\,F0\sin(\omega\,t)$$

$$DEq1 := 100\,\frac{d^2}{dt^2}\,z(t) + 2\,\frac{d}{dt}\,z(t) + \omega^2\,z(t) + (z(t) - y(t))^2 + (z(t) - y(t))^3 + y(t)\,(z(t) - p(t))^2 + (z(t) - p(t))^3 = 0$$

$$DEq2 := 60\,\frac{d^2}{dt^2}\,p(t) + 2\,\frac{d}{dt}\,p(t) + \omega^2\,p(t) + z(t)\,(p(t) - y(t))^2 + (p(t) - y(t))^3 + (p(t) - z(t))^2 + (p(t) - z(t))^3 = 0$$

$$\omega := 10.$$

$$F0 := 20$$

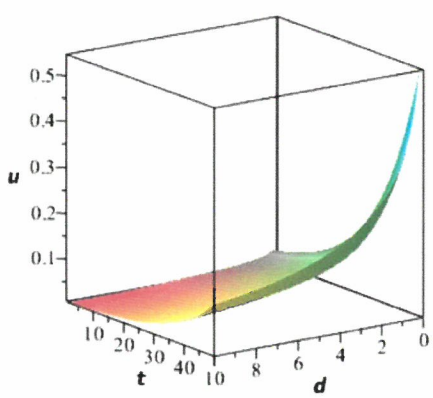
$$\zeta 1 := 0.1$$

$$ini := y(0) = 20, D(y)(0) = 0, z(0) = 0, D(z)(0) = 0, p(0) = 0, D(p)(0) = 0$$

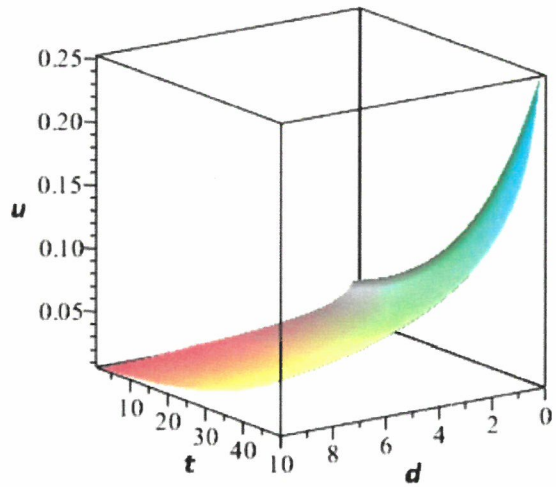
```
> restart : with(DEtools) : with(Student) : with(plots) : non-linear vibration :
>
τ := 100.; DEq0 := 10·diff(y(t), [t$2]) + 2·ζ1·ω·diff(y(t), t) + ω^2·y(t) + (y(t) − z(t))^2
    + (y(t) − z(t))^3 + z(t)·(y(t) − p(t))^2 + (y(t) − p(t))^3 = exp(−t/τ)·(1 − t/τ)·F0
    ·sin(ω·t); DEq1 := 100·diff(z(t), [t$2]) + 2·diff(z(t), t) + ω^2·z(t) + (z(t) − y(t))^2
    + (z(t) − y(t))^3 + y(t)·(z(t) − p(t))^2 + (z(t) − p(t))^3 = 0; DEq2 := 60·diff(p(t), [t
    $2]) + 2·diff(p(t), t) + ω^2·p(t) + z(t)·(p(t) − y(t))^2 + (p(t) − y(t))^3 + (p(t)
    − z(t))^2 + (p(t) − z(t))^3 = 0; ω := 10.; F0 := 20; ζ1 := 0.1; ini := y(t)|_{t=0} = F0,
    D(y)(0) = 0, z(t)|_{t=0} = 0, D(z)(0) = 0, p(t)|_{t=0} = 0, D(p)(0) = 0; F := dsolve({DEq0,
    DEq1, DEq2, ini}, {y(t), z(t), p(t)}, type = numeric); sol2 := dsolve({DEq0, DEq1,
    DEq2, ini}, {y(t), z(t), p(t)}, numeric, output = listprocedure, abserr = 0.001) : T10 :=
    0.0; T2 := 10.; y2 := 25.; pl11 := odeplot(sol2, [t, y(t)], t = T10 ..T2, numpoints = 500,
    labels = ["t", "y(t)"], thickness = 2, title
    = ` Sinusoidal nonlinear biodynamic model vibration`, titlefont = [COURIER, Bold, 14],
    labelfont = [COURIER, BOLDITALIC, 12], color = red, axes = boxed, view = [T10 ..T2,
    -y2 ..y2], legend = ["y(t)"], legendstyle = [font = ["HELVETICA", 9], location = right])
    : pl12 := odeplot(sol2, [t, z(t)], t = T10 ..T2, numpoints = 500, labels = ["t", "z(t)"],
    thickness = 2, title = ` Sinusoidal nonlinear biodynamic model  vibration`, titlefont
    = [COURIER, Bold, 14], labelfont = [COURIER, BOLDITALIC, 12], color = green, axes
    = boxed, view = [T10 ..T2, -y2 ..y2], legend = ["z(t)"], legendstyle = [font
    = ["HELVETICA", 9], location = right]) : pl13 := odeplot(sol2, [t, p(t)], t = T10 ..T2,
    numpoints = 500, labels = ["t", "p(t)"], thickness = 2, title
    = ` Sinusoidal nonlinear biodynamic model vibration`, titlefont = [COURIER, Bold, 14],
    labelfont = [COURIER, BOLDITALIC, 12], color = blue, axes = boxed, view = [T10 ..T2,
    -y2 ..y2], legend = ["p(t)"], legendstyle = [font = ["HELVETICA", 9], location = right])
    : plots[display]({pl11, pl12, pl13});
```

2.8 Flamant Bousinesq

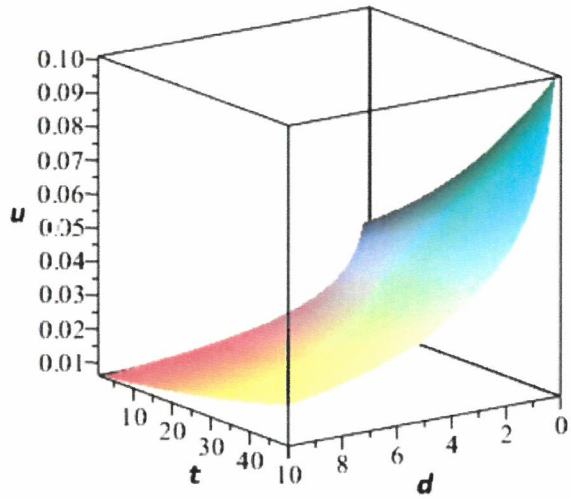
Time dependent Fractional Calculus Viscoelastic Settlement Magnitude



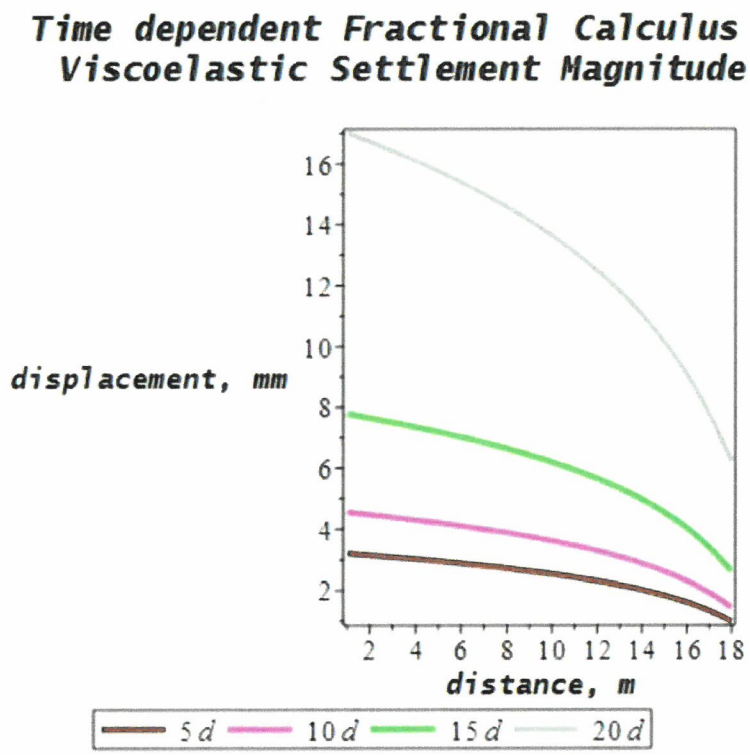
Time dependent Fractional Calculus Viscoelastic Settlement Magnitude

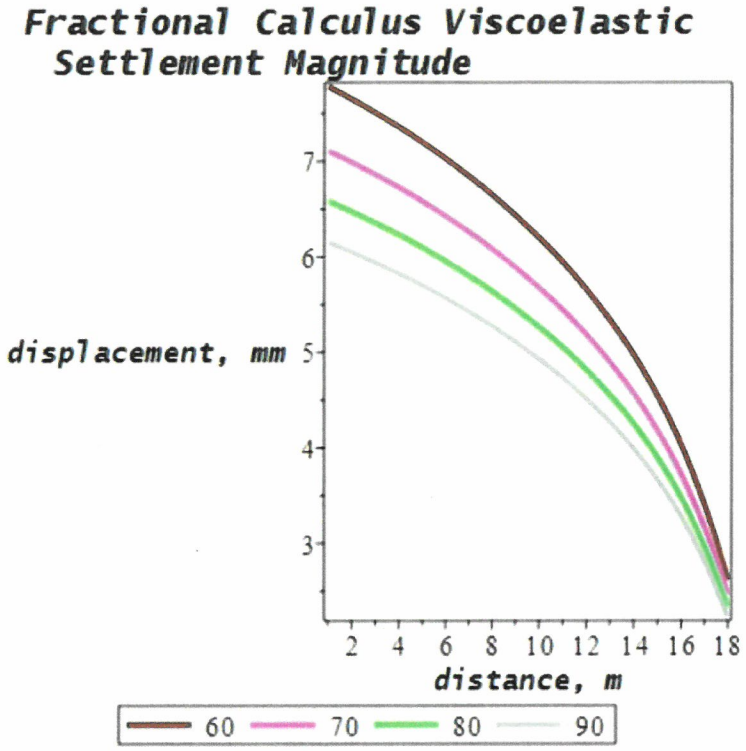


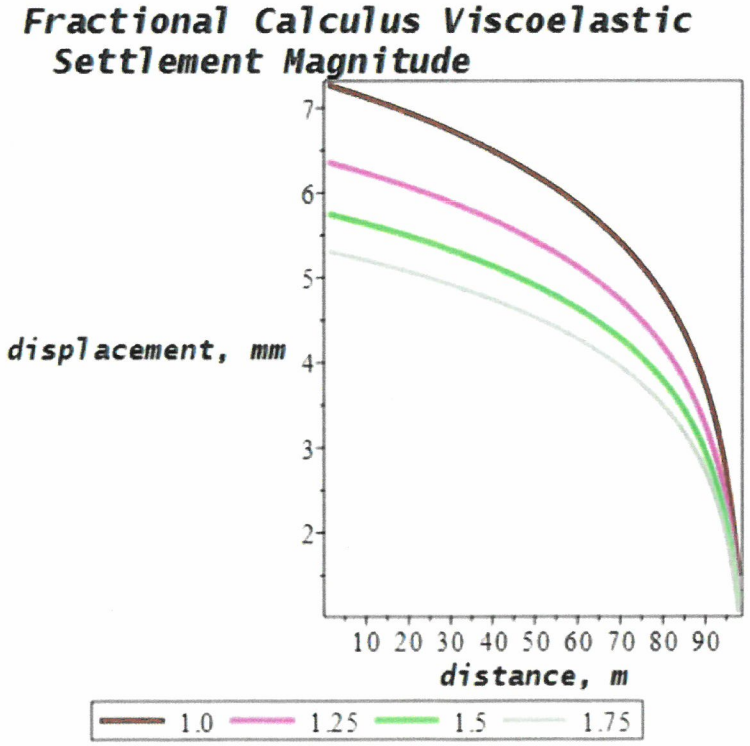
Time dependent Fractional Calculus Viscoelastic Settlement Magnitude



```
restart; with(Student); with(plots); N := 50; p0 := 1; K := 80; G := 60; E := 144; r0 := 15; eta := 1000; alpha := [1., 1.2, 1.7]; tau1 := evalf(4*eta/(3*K+G)); tau2 := evalf(eta/G); for I to 3 do ur[i] := proc (t) options operator, arrow; evalf(9*p0^2*(1-sum(-(t/tau1)^n/GAMMA(n*alpha[i]+1), n = 0 .. N))^2/(12*K+4*G)^2) end proc; `u&theta;`[i] := proc (r, t) options operator, arrow; evalf((1/4)*p0^2*ln(r0/r)^2*((3-3*(sum(-(t/tau1)^n/GAMMA(n*alpha[i]+1), n = 0 .. N)))/(3*K+G)+(1-sum(-(t/tau2)^n/GAMMA(n*alpha[i]+1), n = 0 .. N))/G)^2/3.14159^2) end proc; zz := proc (r, t) options operator, arrow; evalf((ur[i](t)+`u&theta;`[i](r, t))^.5) end proc; plot3d(zz(r, t), r = 0 .. 10, t = 1 .. 50, axes = boxed, title = ` Time dependent Fractional Calculus Viscoelastic Settlement Magnitude  `, titlefont = [COURIER, Bold, 14], labels = ["d", "t", "u"], labelfont = [COURIER, BOLDITALIC, 12]) end do;
```

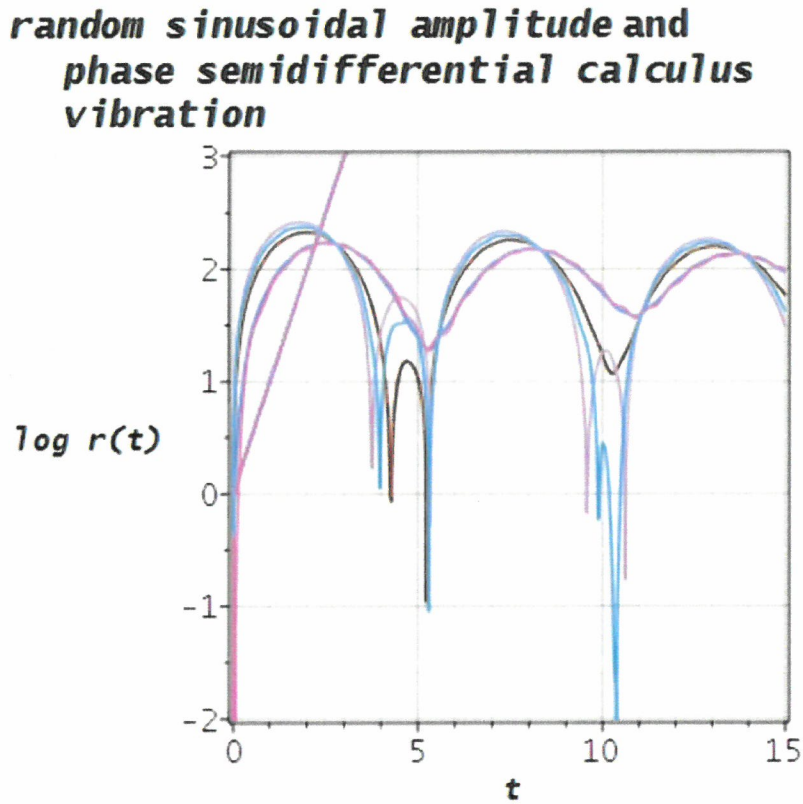



Fractional Calculus Viscoelastic
Settlement Magnitude
displacement, mm
7
6
5
4
3
2 4 6 8 10 12 14 16 18
distance, m
60 70 80 90

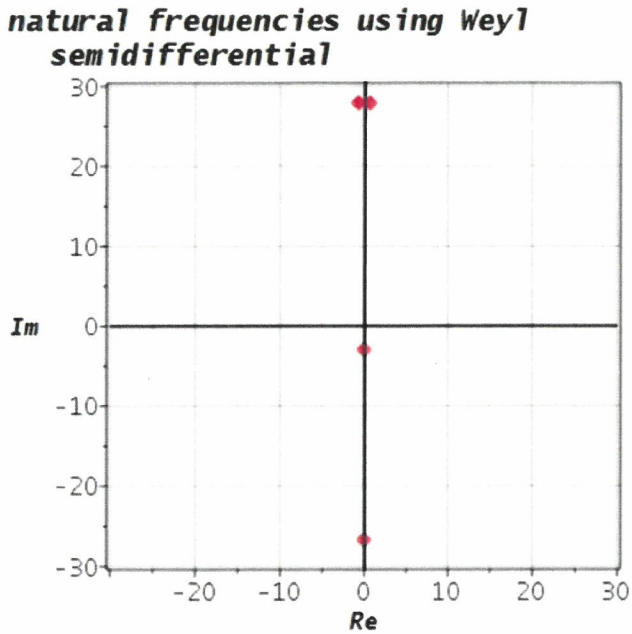

Fractional Calculus Viscoelastic
Settlement Magnitude
displacement, mm
7
6
5
4
3
2
10 20 30 40 50 60 70 80 90
distance, m
1.0 1.25 1.5 1.75

2.9 Fractional Calculus Bubble Oscillator

```
restart; with(Student); with(plots); with(RandomTools); N := 50; p0 := 10^2; K := 80; G := 60; E := 144; r0
:= 1; eta := 1000; alpha := .5; tau1 := evalf(4*eta/(3*K+G)); tau2 := evalf(eta/G); T := 4; col1 :=
seq(COLOR(RGB, rand()/10^11, rand()/10^12, rand()/10^12), j = 1 .. T); t5 := [5, 10, 15, 20]; for I to T do
ur[i] := subs(r0 = 1, p0 = 10^6, K = 80, G = 60, alpha = .6, t1 = t5[i], evalf((-3*p0*(1-sum(-
(t1/tau1)^n/GAMMA(alpha*n+1), n = 0 .. N))/(4*(3*K+G)))^2)); `u&theta;`[i] := proc I options operator,
arrow; subs(r0 = 1, p0 = 1000000, K = 80, G = 60, alpha = .6, t1 = t5[i], evalf((1/4)*p0^2*ln(r0/r)^2*((3-
3*(sum(-(t1/tau1)^n/GAMMA(alpha*n+1), n = 0 .. N)))/(3*K+G)+(1-sum(-
(t1/tau2)^n/GAMMA(alpha*n+1), n = 0 .. N))/G)^2/3.14159^2)) end proc; zz1[i] := proc I options
operator, arrow; evalf((ur[i]+`u&theta;`[i]I)^.5) end proc; R := 20; pl[i] := plot([zz1[i](R-r)], r = 1 .. R-2,
axes = boxed, title = ` Time dependent Fractional Calculus Viscoelastic Settlement Magnitude `, titlefont
= [COURIER, Bold, 14], labels = ["distance, m", "displacement, mm"], labelfont = [COURIER, BOLDITALIC,
12], color = col1[i], legend = [t5[i]*d], style = line, thickness = 3) end do; plots[display]([seq(pl[i], I = 1 ..
T)]); G1 := [60, 70, 80, 90]; for I to T do ur2[i] := subs(r0 = 1, p0 = 10^6, K = 80, alpha = .6, t1 = t5[3],
evalf((-3*p0*(1-sum(-(t1/tau1)^n/GAMMA(alpha*n+1), n = 0 .. N))/(4*(3*K+G1[i])))^2)); `u&theta;2`[i]
:= proc I options operator, arrow; subs(r0 = 1, p0 = 1000000, K = 80, alpha = .6, t1 = t5[3],
evalf((1/4)*p0^2*ln(r0/r)^2*((3-3*(sum(-(t1/tau1)^n/GAMMA(alpha*n+1), n = 0 .. N)))/(3*K+G1[i])+(1-
sum(-(t1/tau2)^n/GAMMA(alpha*n+1), n = 0 .. N))/G1[i])^2/3.14159^2)) end proc; zz2[i] := proc I
options operator, arrow; evalf((ur2[i]+`u&theta;2`[i]I)^.5) end proc; R := 20; pl2[i] := plot([zz2[i](R-r)], r =
1 .. R-2, axes = boxed, title = ` Fractional Calculus Viscoelastic Settlement Magnitude `, titlefont =
[COURIER, Bold, 14], labels = ["distance, m", "displacement, mm"], labelfont = [COURIER, BOLDITALIC,
12], color = col1[i], legend = [G1[i]], style = line, thickness = 3) end do; plots[display]([seq(pl2[i], I = 1 ..
T)]); r01 := [1, 5, 10, 15]; for I to T do ur3[i] := subs(r0 = r01[i], p0 = 10^6, K = 80, alpha = .6, t1 = t5[3],
evalf((-3*p0*(1-sum(-(t1/tau1)^n/GAMMA(alpha*n+1), n = 0 .. N))/(4*(3*K+G1[1])))^2)); `u&theta;3`[i]
:= proc I options operator, arrow; subs(r0 = r01[i], p0 = 1000000, K = 80, alpha = .6, t1 = t5[3],
evalf((1/4)*p0^2*ln(r0/r)^2*((3-3*(sum(-(t1/tau1)^n/GAMMA(alpha*n+1), n = 0 .. N)))/(3*K+G1[1])+(1-
sum(-(t1/tau2)^n/GAMMA(alpha*n+1), n = 0 .. N))/G1[1])^2/3.14159^2)) end proc; zz3[i] := proc I
options operator, arrow; evalf((ur3[i]+`u&theta;3`[i]I)^.5) end proc; R := 100; pl3[i] := plot([zz3[i](R-r)], r
= 1 .. R-2, axes = boxed, title = ` Fractional Calculus Viscoelastic Settlement Magnitude `, titlefont =
[COURIER, Bold, 14], labels = ["distance, m", "displacement, mm"], labelfont = [COURIER, BOLDITALIC,
12], color = col1[i], legend = [r01[i]], style = line, thickness = 3) end do; plots[display]([seq(pl3[i], I = 1 ..
T)]); alpha1 := [1.0, 1.25, 1.5, 1.75]; for I to T do ur4[i] := subs(r0 = r01[1], p0 = 10^6, K = 80, alpha =
alpha1[i], t1 = t5[3], evalf((-3*p0*(1-sum(-(t1/tau1)^n/GAMMA(n*alpha1[i]+1), n = 0 ..
N))/(4*(3*K+G1[1])))^2)); `u&theta;4`[i] := proc I options operator, arrow; subs(r0 = r01[1], p0 =
1000000, K = 80, alpha = alpha1[i], t1 = t5[3], evalf((1/4)*p0^2*ln(r0/r)^2*((3-3*(sum(-
(t1/tau1)^n/GAMMA(n*alpha1[i]+1), n = 0 .. N)))/(3*K+G1[1])+(1-sum(-
(t1/tau2)^n/GAMMA(n*alpha1[i]+1), n = 0 .. N))/G1[1])^2/3.14159^2)) end proc; zz4[i] := proc I options
operator, arrow; evalf((ur4[i]+`u&theta;4`[i]I)^.5) end proc; R := 100; pl4[i] := plot([zz4[i](R-r)], r = 1 .. R-
2, axes = boxed, title = ` Fractional Calculus Viscoelastic Settlement Magnitude `, titlefont = [COURIER,
Bold, 14], labels = ["distance, m", "displacement, mm"], labelfont = [COURIER, BOLDITALIC, 12], color =
col1[i], legend = [alpha1[i]], style = line, thickness = 3) end do; plots[display]([seq(pl4[i], I = 1 .. T)]);
```



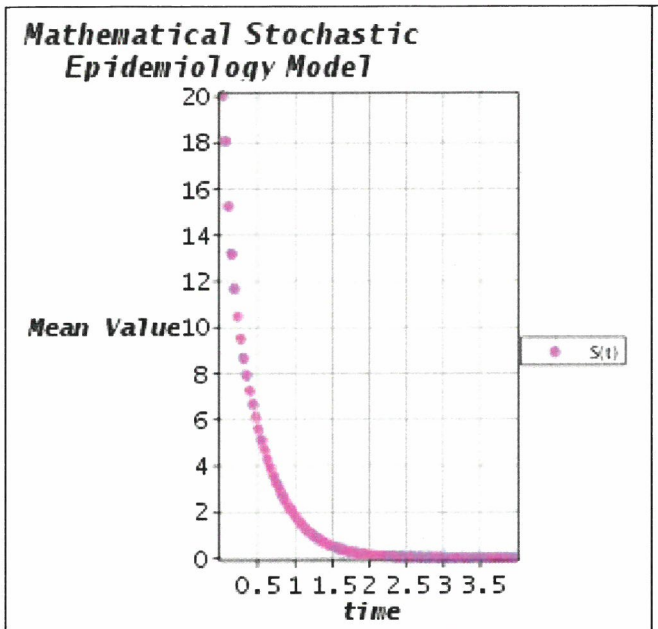
restart; with(Student); with(inttrans); with(RandomTools); n := 5; A := Generate(list(float(range = .1 .. 10., method = uniform), n)); Pie := evalf(Pi); phi := Generate(list(float(range = -Pie .. Pie, method = uniform), n)); col1 := seq(COLOR(RGB, rand()/10^12, rand()/10^12, rand()/10^12), I = 1 .. n); assume(0 < a); omega1 := 10.; alpha1 := .5; mu0 := 0.1e-2; rho0 := 1.0*10^3; R0 := 5.0*10^(-3); a0 := 10.; sigma1 := 72.8*10^(-3); p0 := 1.983*10^(-5); A1 := evalf(4*mu0/(rho0*R0^2)); B1 := evalf(4*sigma1/(rho0*omega1*R0^2)+2*p0*R0*(1/(rho0*omega1*R0^2))); tau := 100.; di1 := fracdiff(y(t), t, -.5); di0 := laplace(di1, t, s); ic1 := invlaplace(di0, s, 0); sol3 := solve(s^2+A1*s^alpha1+B1 = 0, s); y1 := 3.; for I to n do ic0[i] := ((a0*A[i]*sin(omega1*0+phi[i])*Heaviside(t)-p0)*R0-2*sigma1)/(rho0*omega1*R0^3); numerator := proc (s) options operator, arrow; laplace(((a0*A[i]*sin(omega1*t+phi[i])*Heaviside(t)-p0)*R0-2*sigma1)/(rho0*omega1*R0^3), t, s) end proc; paf := proc (s) options operator, arrow; convert(1/((s-sol3[1])*(s-sol3[2])), parfrac) end proc; fls[i] := invlaplace(paf(s)*(numerator(s)+ic1+ic0[i]), s, t); plt[i] := plot([t, log10(abs(fls[i](t)))], t = 0 .. 15, -2 .. y1, axes = boxed, style = line, color = col1[i], gridlines, title = 'random*sinusoidal*amplitude and phase*semidifferential*calculus*vibration', titlefont = [COURIER, Bold, 14], labels = ["t", "log r(t)"], labelfont = [COURIER, BOLDITALIC, 12], axesfont = [COURIER, 12]) end do; plots[display]({seq(plt[i], I = 1 .. n)});

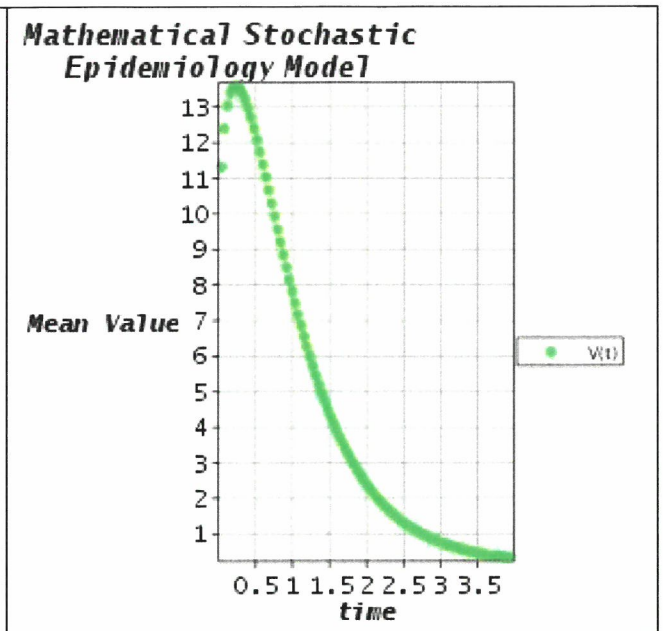


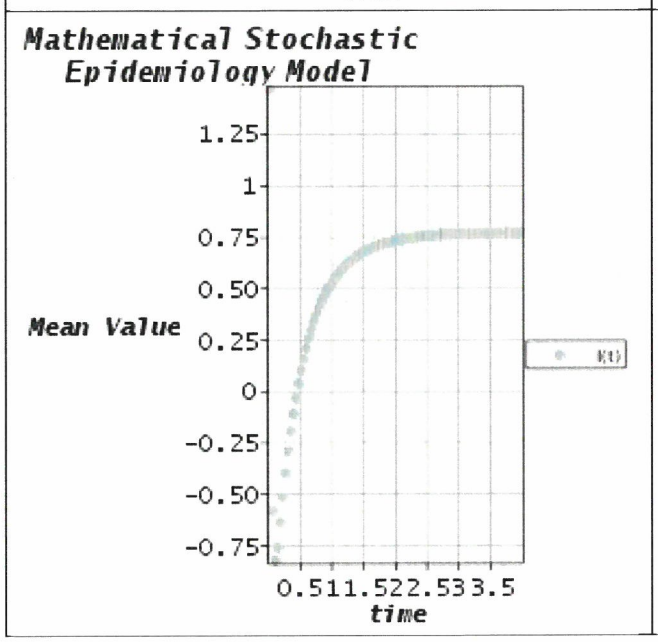

```
restart; with(Student); with(LinearAlgebra); with(DETools); with(plots); with(RandomTools); n := 2; col1
:= seq(COLOR(RGB, rand()/10^12, rand()/10^12, rand()/10^12), j = 1 .. n); A0 := Matrix(2, 2, [[80.81722,
2.31941332208709], [2.31941332208709, .29784188199686]]); g := 9.81; v := 3.5; omega1 := 10; A := v .
Matrix(2, 2, [[0, 33.86641391492494], [-.85035641456978, 1.68540397397560]]); K0 := Matrix(2, 2, [[-
80.95, -2.59951685249872], [-2.59951685249872, -.80329488458618]]); K2 := Matrix(2, 2, [[0,
76.59734589573222], [0, 2.65431523794604]]); B := evalf(K2*v^2+K0*g); q := Matrix(2, 1, [x(t), y(t)]);
zer0 := Matrix(2, 1, [0, 0]); Id1 := IdentityMatrix(2); alpha := .5; de0 := A0 . (diff(q, t, t))+A . (diff(q, t))+B .
q; ampl := Matrix(2, 2, [[a0, 0], [0, b0]]); ampl2 := Matrix(2, 2, [[a^alpha, 0], [0, b^alpha]]); qs := Matrix(2,
1, [exp(a*t), exp(b*t)]); q2 := ampl . qs; q2f := ampl2 . ampl . qs; diff(qs, t, t); de1 := diff(q2, t, t)+B . q2;
sys0 := de1+A . q2f; sys1 := simplify(sys0[1]); sys2 := simplify(sys0[2]); sys1f := collect(factor(sys1), [a0,
b0]); sys2f := collect(factor(sys2), [a0, b0]); m11 := coeff(sys1f, a0); m21 := coeff(sys2f, a0); m12 :=
coeff(sys1f, b0); m22 := coeff(sys2f, b0); t := 0.; mat := Matrix(2, 2, [[m11, m12], [m21, m22]]); a :=
I*omega; b := I*omega; poly := Determinant(mat); dl := 1.; natff := solve(poly = 0, omega); for I to 4 do
pl[i] := complexplot([natff[i]], -30 .. 30, -30 .. 30, style = point, symbol = soliddiamond, symbolsize = 16,
color = red, gridlines, axes = boxed, title = 'natural*frequencies*using*Weyl*semidifferential', titlefont =
[COURIER, Bold, 14], labels = ["Re", "Im"], labelfont = [COURIER, BOLDITALIC, 12], axesfont = [COURIER,
12]) end do; g := proc (x) options operator, arrow; 0*x end proc; h := proc (x) options operator, arrow;
10000000*x end proc; plx := plot(g(x), x = -30 .. 30, y = -30 .. 30, style = line, thickness = 2, color = black);
ply := plot(h(x), x = -30 .. 30, y = -30 .. 30, style = line, thickness = 2, color = black); plt :=
plots[display](seq(pl[i], I = 1 .. 4), plx, ply);
```

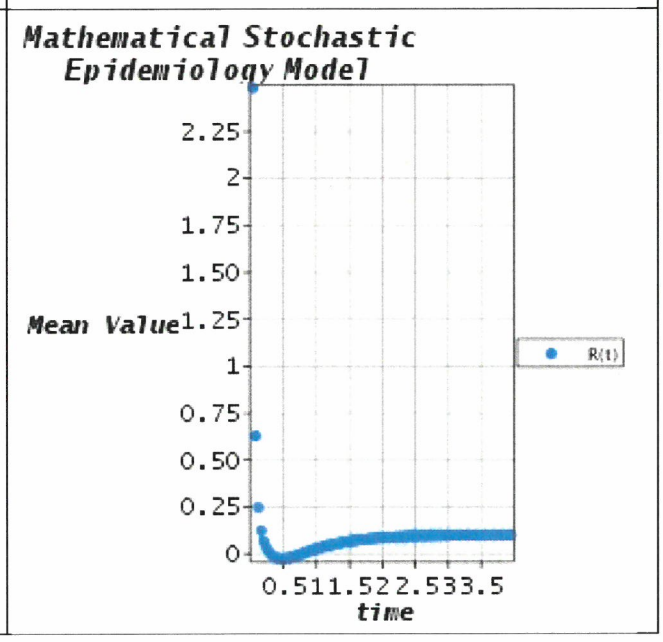
3.0 Epidemiological Model

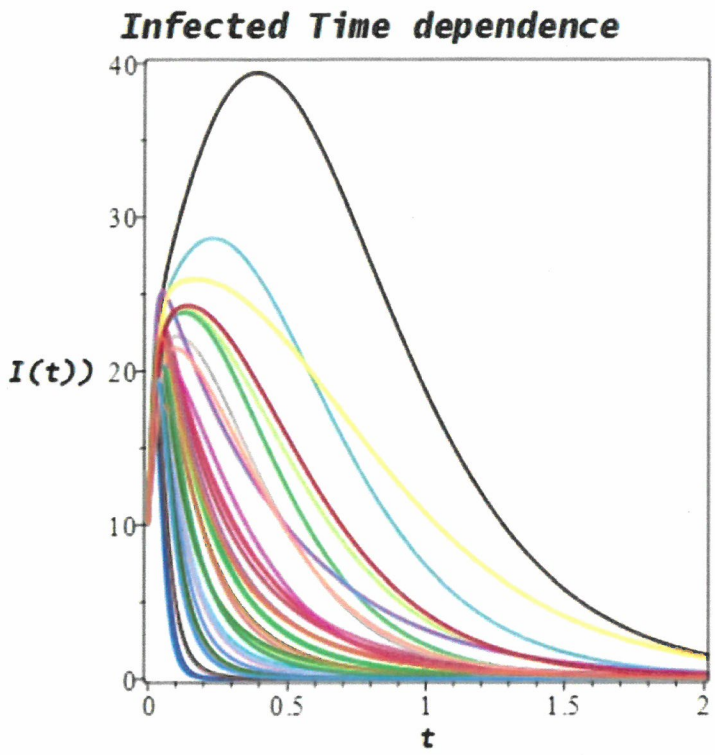
```
restart : with(DEtools) : with(Student) : with(plots) : with(Statistics) : with(RandomTools) :
with(Threads) : n := 35; col10 := seq(COLOR(RGB, rand()/10^12, rand()/10^12, rand()/10^12), j
= 1 ..n) : θr := Generate(list(float( range = 0.2 ..1.1, method = uniform), n)) : β1r :=
Generate(list(float( range = 1 ..3, method = uniform), n)) : γr :=
Generate(list(float( range = 2 ..3, method = uniform), n)); γ0r :=
Generate(list(float( range = 2 ..4, method = uniform), n)) : β0r :=
Generate(list(float( range = 1 ..4, method = uniform), n)) ; μr := Generate(list(float( range
= 1 ..3, method = uniform), n)) ;for i from 1 to n do ; de1 := d/dt s1(t) − μr[i] + μr[i]
·s1(t) + β0r[i]·v1(t)·I1(t) + θr[i]·s1(t) = 0; de2 := d/dt v1(t) − θr[i]·s1(t) + β1r[i]
·v1(t)·I1(t) + γr[i]·v1(t) + μr[i]·v1(t) = 0; de3 := d/dt I1(t) − β0r[i]·s1(t)·I1(t)
− β1r[i]·v1(t)·I1(t) + γr[i]·I1(t) + μr[i]·I1(t) = 0; de4 := d/dt R11(t) − γr[i]·v1(t)
− γr[i]·I1(t) + μr[i]·R11(t) = 0; sol1[i] := dsolve({de1, de2, de3, de4, s1(0) = 10,
v1(0) = 10, I1(0) = 10, R11(0) = 10}, {s1(t), v1(t), I1(t), R11(t)}, numeric, output
= listprocedure, abserr = 0.001) : T10 := 0.0; T2 := 4; pl11[i] := odeplot(sol1[i], [t,
I1(t)], t = T10 ..T2, numpoints = 500, labels = ["t", "I(t))"], thickness = 2, title
= ` Infected Time dependence`, titlefont = [COURIER, Bold, 14], labelfont = [COURIER,
BOLDITALIC, 12], color = col10[i], axes = boxed, view = [T10 ..2, 0 ..40]) : pl21[i] :=
odeplot(sol1[i], [[t, R11(t)]], t = T10 ..T2, numpoints = 500, labels = ["t", " R(t)"],
thickness = 2, title = `Recovered Time dependence `, titlefont = [COURIER, Bold, 14],
labelfont = [COURIER, BOLDITALIC, 12], color = col10[i], axes = boxed, view = [T10
..T2, 0 ..40]) : pl31[i] := odeplot(sol1[i], [[t, v1(t)]], t = T10 ..T2, numpoints = 500,
labels = ["t", " V(t)"], thickness = 2, title = `Vaccine density Time dependence `, titlefont
= [COURIER, Bold, 14], labelfont = [COURIER, BOLDITALIC, 12], color = col10[i],
axes = boxed, view = [T10 ...2, 0 ..10]) :end do : q[1] := plots[display](seq(pl11[i], i = 1
.. n)); q[2] := plots[display](seq(pl21[i], i = 1 ..n)); q[3] :=
plots[display](seq(pl31[i], i = 1 ..n)); nint := 100; nn := nint − 1; np := (T2 − T10)/nint;
for w from 1 to 5 do for j from 1 to n do sol2[j] := [seq(sol1[j][w](T10 + i·np), i = 1
..nn)]; end do;for m from 1 to n do v11[m] := [ seq(rhs(sol2[m][h]), h = 1 ..nn)] ;end
do; v10 := [seq(Add(v11[m][j], m = 1 ..n), j = 1 ..nn)]; T2 := [seq(T10 + i·np, i = 1
..nn)]; pts := seq([T2[i], v10[i]/n], i = 1 ..nn); Titles := ["Time", "S(t)", "V(t)", "I(t)",
"R(t)"];; pll[w] := pointplot([pts], axes = boxed, color = col10[w], symbol = solidcircle,
symbolsize = 13, title ='Mathematical Stochastic Epidemiology Model', titlefont
= [COURIER, Bold, 14], labels = ["time", "Mean Value"], labelfont = [COURIER,
BOLDITALIC, 12], axesfont = [COURIER, 12], gridlines, tickmarks = [10, 10], legend
= Titles[w], legendstyle = [font = ["HELVETICA", 9], location = right]);end do:
m1[1] := plots[display]([seq(pll[w], w = 2 ..5)]); display(Array(1 ..2, 1 ..2, [[pll[2],
pll[3]], [pll[4], pll[5]]])); display(Array(1 ..2, 1 ..2, [[q[1], q[2]], [q[3], m1[1]]]));
```

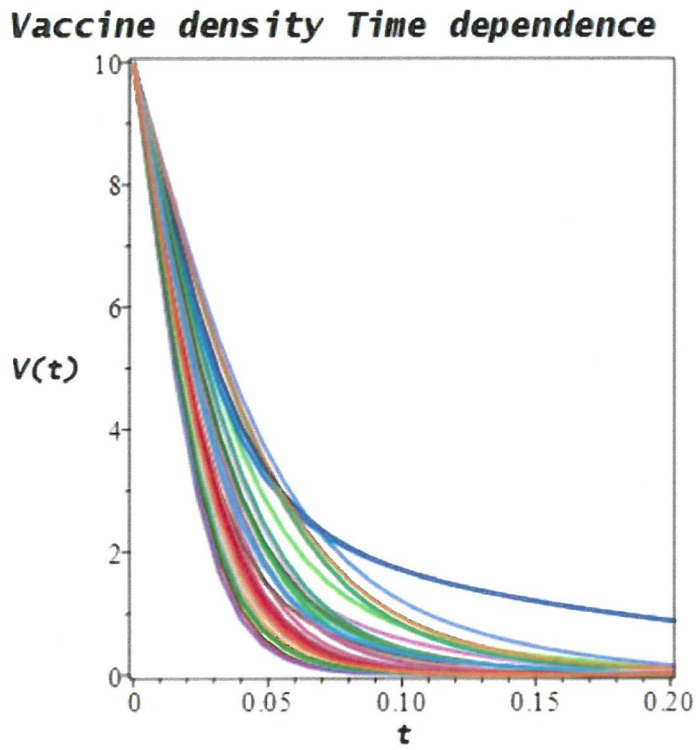

Mathematical Stochastic
Epidemiology Model
Mean Value
time
S(t)

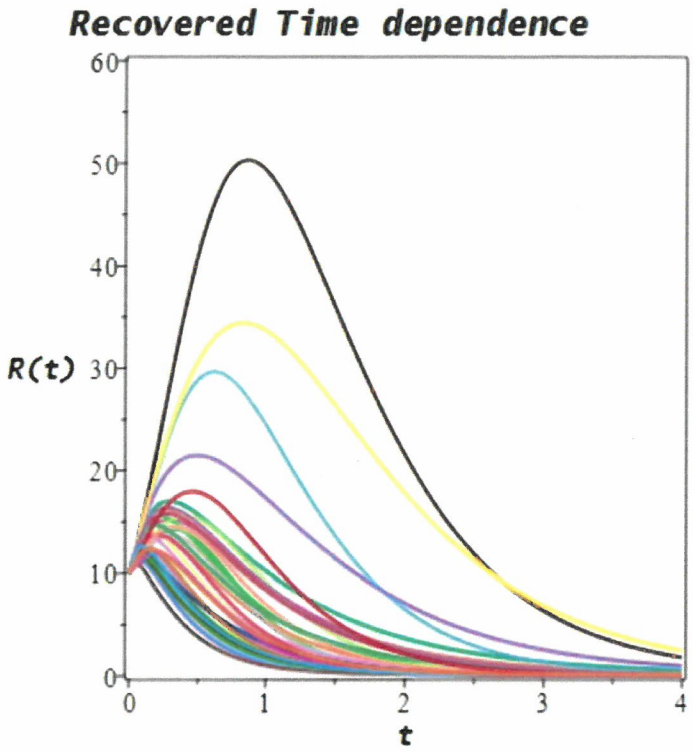

Mathematical Stochastic
Epidemiology Model
Mean Value
time
V(t)

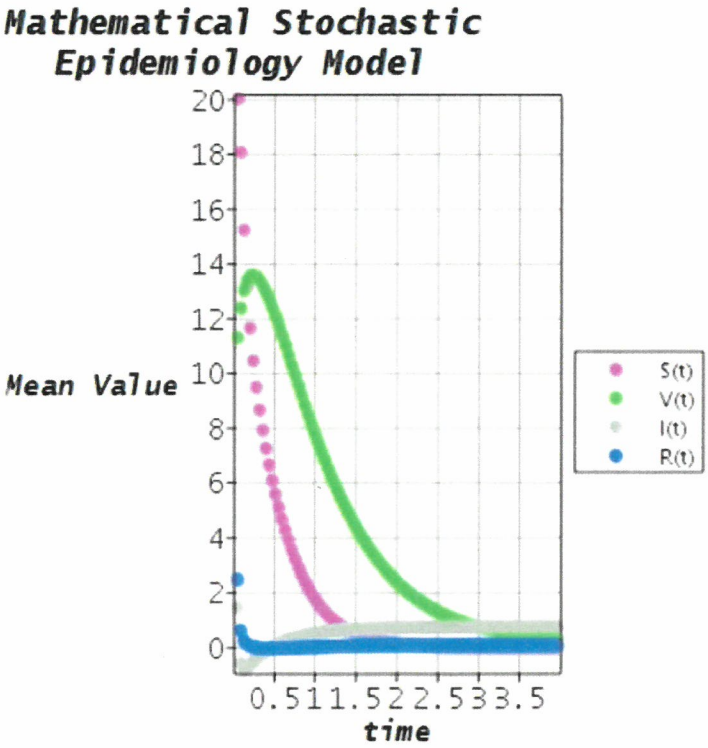

Mathematical Stochastic
Epidemiology Model
Mean Value
time
I(t)


Mathematical Stochastic
Epidemiology Model
Mean Value
time
R(t)


Infected Time dependence
I(t))
t
0
10
20
30
40
0.5
1
1.5
2


Vaccine density Time dependence
V(t)
t
0
2
4
6
8
10
0.05
0.10
0.15
0.20


Recovered Time dependence
R(t)
t
0
10
20
30
40
50
60
0
1
2
3
4


Mathematical Stochastic
Epidemiology Model
Mean Value
time
S(t)
V(t)
I(t)
R(t)
0
2
4
6
8
10
12
14
16
18
20
0.5
1
1.5
2
2.5
3
3.5

3.1 Pi Thereom Parameters: Falling Particle

```
restart; with(Student); with(combinat, randcomb); with(linalg); with(Student[LinearAlgebra]);
with(plots); m := 10; n1 := 3; dd := m-n1; N := factorial(m)/(factorial(m-n1)*factorial(n1)); m0 := [delta, L,
A, rho, U, c, p, mu, G, F]; for j to N do mc[j] := randcomb(m0, n1) end do; M := [seq(mc[k], k = 1 .. N)]; for
k to N do m0s[k] := convert(m0, set); Mc[k] := convert(M[k], set); A0[k] := m0s[k] minus Mc[k]; Cm[k] :=
[A0[k], Mc[k]]; V1[k] := Vector(dd, [seq(A0[k][j], j = 1 .. dd)]); V2[k] := Vector(n1, [seq(Mc[k][j], j = 1 ..
n1)]); V1tr[k] := Transpose(V1[k]); V2t[k] := Transpose(V2[k]) end do; Am2[1] := [seq(A0[1][i], I = 1 .. dd)];
delta := Vector(3, [0, 1, 0]); L := Vector(3, [0, 1, 0]); A := Vector(3, [0, 2, 0]); rho := Vector(3, [1, -3, 0]); U
:= Vector(3, [0, 1, -1]); c := Vector(3, [0, 1, -1]); p := Vector(3, [1, -1, -2]); mu := Vector(3, [1, -1, -1]); G :=
Vector(3, [1, -2, -2]); F := Vector(3, [1, 1, -2]); for k to N do AM1[k] := Matrix(n1, dd, [seq(V1tr[k][i], I = 1 ..
dd)]); Bm1[k] := Matrix(n1, n1, [seq(V2t[k][i], I = 1 .. n1)]); det1[k] := Determinant(Bm1[k]) end do; for n
to N do if det1[n] = 0 then next end if; Bm1i[n] := inverse(Bm1[n]); I01[n] := -1.*Matrix(n1, n1, shape =
identity); A1[n] := I01[n] . AM1[n]; Dim[n] := Bm1i[n] . A1[n] end do; for n to N do if det1[n] = 0 then next
end if; dimpar[n] := seq(V2t[n](1, 1)^convert(Dim[n](1, i), rational)*V2t[n](1, 2)^convert(Dim[n](2, i),
rational)*V2t[n](1, 3)^convert(Dim[n](3, i), rational)*V1tr[n](1, i), I = 1 .. dd) end do;
```

(boundary layer thickness, length, area, density, speed, sound speed, pressure, shear viscosity, pressure gradient, force-drag)

$$m0 := [\delta, L, A, \rho, U, c, p, \mu, G, F]$$

$$dimpar_1 := \frac{pA}{F}, \frac{\sqrt{F}\,G}{p^{3/2}}, \frac{\sqrt{p}\,L}{\sqrt{F}}, \frac{U}{c}, \frac{\sqrt{p}\,\delta}{\sqrt{F}}, \frac{c\mu}{\sqrt{F}\sqrt{p}}, \frac{c^2\rho}{p}$$

$$dimpar_2 := \frac{A}{L^2}, \frac{F}{L^2 c^2 \rho}, \frac{LG}{c^2\rho}, \frac{U}{c}, \frac{\delta}{L}, \frac{\mu}{Lc\rho}, \frac{p}{c^2\rho}$$

$$dimpar_3 := \frac{A}{\delta^2}, \frac{F}{U\delta\mu}, \frac{\delta^2 G}{U\mu}, \frac{L}{\delta}, \frac{c}{U}, \frac{\delta p}{U\mu}, \frac{U\delta\rho}{\mu}$$

$$dimpar_4 := \frac{A}{L^2}, \frac{L^3 G}{F}, \frac{c}{U}, \frac{\delta}{L}, \frac{LU\mu}{F}, \frac{L^2 p}{F}, \frac{L^2 U^2 \rho}{F}$$

$$dimpar_5 := \frac{A}{L^2}, \frac{\rho F}{\mu^2}, \frac{L^3 \rho G}{\mu^2}, \frac{L\rho U}{\mu}, \frac{L\rho c}{\mu}, \frac{\delta}{L}, \frac{L^2 \rho p}{\mu^2}$$

$$dimpar_7 := \frac{A}{L^2}, \frac{L^3 G}{F}, \frac{LU\mu}{F}, \frac{L\mu c}{F}, \frac{\delta}{L}, \frac{L^2 p}{F}, \frac{\rho F}{\mu^2}$$

$$dimpar_8 := \frac{A}{\delta^2}, \frac{F}{G\delta^3}, \frac{L}{\delta}, \frac{U}{c}, \frac{c\mu}{G\delta^2}, \frac{p}{G\delta}, \frac{c^2\rho}{G\delta}$$

$$dimpar_9 := \frac{F}{Ap}, \frac{\sqrt{A}\,G}{p}, \frac{L}{\sqrt{A}}, \frac{c}{U}, \frac{\delta}{\sqrt{A}}, \frac{U\mu}{\sqrt{A}\,p}, \frac{U^2\rho}{p}$$

$$dimpar_{10} := \frac{A^{3/2} G}{F}, \frac{L}{\sqrt{A}}, \frac{\sqrt{A}\,\mu U}{F}, \frac{\sqrt{A}\,\mu c}{F}, \frac{\delta}{\sqrt{A}}, \frac{pA}{F}, \frac{\rho F}{\mu^2}$$

$$dimpar_{12} := \frac{A}{L^2}, \frac{L^3 G}{F}, \frac{c}{U}, \frac{\delta}{L}, \frac{L U \mu}{F}, \frac{L^2 p}{F}, \frac{L^2 U^2 \rho}{F}$$

$$dimpar_{13} := \frac{A}{L^2}, \frac{\rho F}{\mu^2}, \frac{L^3 \rho G}{\mu^2}, \frac{L \rho U}{\mu}, \frac{L \rho c}{\mu}, \frac{\delta}{L}, \frac{L^2 \rho p}{\mu^2}$$

$$dimpar_{14} := \frac{G^2 A}{p^2}, \frac{G^2 F}{p^3}, \frac{G L}{p}, \frac{c}{U}, \frac{G \delta}{p}, \frac{G U \mu}{p^2}, \frac{U^2 \rho}{p}$$

$$dimpar_{15} := \frac{A^{3/2} G}{F}, \frac{L}{\sqrt{A}}, \frac{\sqrt{A}\sqrt{\rho}\, U}{\sqrt{F}}, \frac{\sqrt{A}\sqrt{\rho}\, c}{\sqrt{F}}, \frac{\delta}{\sqrt{A}}, \frac{\mu}{\sqrt{F}\sqrt{\rho}}, \frac{p A}{F}$$

$$dimpar_{16} := \frac{G^{2/3} A}{F^{2/3}}, \frac{G^{1/3} L}{F^{1/3}}, \frac{\sqrt{\rho}\, U}{F^{1/6} G^{1/3}}, \frac{\sqrt{\rho}\, c}{F^{1/6} G^{1/3}}, \frac{G^{1/3} \delta}{F^{1/3}}, \frac{G^{\frac{1}{9007199254740992}} \mu}{\sqrt{F}\sqrt{\rho}},$$

$$\frac{p}{F^{1/3} G^{2/3}}$$

$$dimpar_{17} := \frac{G^2 A}{p^2}, \frac{G^2 F}{p^3}, \frac{G L}{p}, \frac{G U \mu}{p^2}, \frac{G \mu c}{p^2}, \frac{G \delta}{p}, \frac{p^3 \rho}{G^2 \mu^2}$$

$$dimpar_{19} := \frac{A}{L^2}, \frac{F}{L^2 p}, \frac{G L}{p}, \frac{\sqrt{\rho}\, U}{\sqrt{p}}, \frac{\sqrt{\rho}\, c}{\sqrt{p}}, \frac{\delta}{L}, \frac{\mu}{L\sqrt{p}\sqrt{\rho}}$$

$$dimpar_{20} := \frac{A}{\delta^2}, \frac{F}{c \delta \mu}, \frac{\delta^2 G}{c \mu}, \frac{L}{\delta}, \frac{U}{c}, \frac{\delta p}{c \mu}, \frac{c \delta \rho}{\mu}$$

$$dimpar_{21} := \frac{A}{\delta^2}, \frac{F}{\delta^2 p}, \frac{G \delta}{p}, \frac{L}{\delta}, \frac{\sqrt{\rho}\, U}{\sqrt{p}}, \frac{\sqrt{\rho}\, c}{\sqrt{p}}, \frac{\mu}{\delta\sqrt{p}\sqrt{\rho}}$$

$$dimpar_{22} := \frac{A}{L^2}, \frac{F}{G L^3}, \frac{c}{U}, \frac{\delta}{L}, \frac{U \mu}{G L^2}, \frac{p}{G L}, \frac{U^2 \rho}{G L}$$

$$dimpar_{23} := \frac{F}{A^{3/2} G}, \frac{L}{\sqrt{A}}, \frac{\sqrt{\rho}\, U}{A^{1/4}\sqrt{G}}, \frac{\sqrt{\rho}\, c}{A^{1/4}\sqrt{G}}, \frac{\delta}{\sqrt{A}}, \frac{\mu}{A^{3/4}\sqrt{G}\sqrt{\rho}}, \frac{p}{\sqrt{A}\, G}$$

$$dimpar_{24} := \frac{c^2 \rho A}{F}, \frac{\sqrt{F}\, G}{c^3 \rho^{3/2}}, \frac{c\sqrt{\rho}\, L}{\sqrt{F}}, \frac{U}{c}, \frac{c\sqrt{\rho}\, \delta}{\sqrt{F}}, \frac{\mu}{\sqrt{F}\sqrt{\rho}}, \frac{p}{c^2 \rho}$$

$$dimpar_{25} := \frac{A}{L^2}, \frac{F}{L^2 p}, \frac{G L}{p}, \frac{\mu U}{L p}, \frac{\mu c}{L p}, \frac{\delta}{L}, \frac{L^2 \rho p}{\mu^2}$$

$$dimpar_{28} := \frac{c^2 \mu^2 A}{F^2}, \frac{F^2 G}{c^3 \mu^3}, \frac{L \mu c}{F}, \frac{U}{c}, \frac{c \mu \delta}{F}, \frac{F p}{c^2 \mu^2}, \frac{\rho F}{\mu^2}$$

$$dimpar_{29} := \frac{A}{L^2}, \frac{F}{L c \mu}, \frac{L^2 G}{c \mu}, \frac{U}{c}, \frac{\delta}{L}, \frac{L p}{c \mu}, \frac{L \rho c}{\mu}$$

$$dimpar_{30} := \frac{p \rho A}{\mu^2}, \frac{\rho F}{\mu^2}, \frac{\mu G}{p^{3/2}\sqrt{\rho}}, \frac{\sqrt{p}\sqrt{\rho}\, L}{\mu}, \frac{\sqrt{\rho}\, U}{\sqrt{p}}, \frac{\sqrt{\rho}\, c}{\sqrt{p}}, \frac{\sqrt{p}\sqrt{\rho}\, \delta}{\mu}$$

$$dimpar_{31} := \frac{G^{2/3} A}{F^{2/3}}, \frac{G^{1/3} L}{F^{1/3}}, \frac{\sqrt{\rho}\, U}{F^{1/6} G^{1/3}}, \frac{\sqrt{\rho}\, c}{F^{1/6} G^{1/3}}, \frac{G^{1/3} \delta}{F^{1/3}}, \frac{G^{\frac{1}{9007199254740992}} \mu}{\sqrt{F}\sqrt{\rho}},$$

$$\frac{p}{F^{1/3} G^{2/3}}$$

$$dimpar_{34} := \frac{A}{\delta^2}, \frac{F}{c^2 \delta^2 \rho}, \frac{\delta G}{c^2 \rho}, \frac{L}{\delta}, \frac{U}{c}, \frac{\mu}{c \delta \rho}, \frac{p}{c^2 \rho}$$

$$dimpar_{37} := \frac{A^{3/2} G}{F}, \frac{L}{\sqrt{A}}, \frac{\sqrt{A}\sqrt{\rho}\, U}{\sqrt{F}}, \frac{\sqrt{A}\sqrt{\rho}\, c}{\sqrt{F}}, \frac{\delta}{\sqrt{A}}, \frac{\mu}{\sqrt{F}\sqrt{\rho}}, \frac{pA}{F}$$

$$dimpar_{38} := \frac{p^2 A}{c^2 \mu^2}, \frac{Fp}{c^2 \mu^2}, \frac{G \mu c}{p^2}, \frac{Lp}{c \mu}, \frac{U}{c}, \frac{\delta p}{c \mu}, \frac{c^2 \rho}{p}$$

$$dimpar_{39} := \frac{A}{\delta^2}, \frac{F}{G \delta^3}, \frac{L}{\delta}, \frac{U}{c}, \frac{c \mu}{G \delta^2}, \frac{p}{G \delta}, \frac{c^2 \rho}{G \delta}$$

$$dimpar_{40} := \frac{A}{L^2}, \frac{F}{G L^3}, \frac{c}{U}, \frac{\delta}{L}, \frac{U \mu}{G L^2}, \frac{p}{G L}, \frac{U^2 \rho}{G L}$$

$$dimpar_{42} := \frac{F}{A^{3/2} G}, \frac{L}{\sqrt{A}}, \frac{\mu U}{A G}, \frac{\mu c}{A G}, \frac{\delta}{\sqrt{A}}, \frac{p}{\sqrt{A}\, G}, \frac{A^{3/2} G \rho}{\mu^2}$$

$$dimpar_{43} := \frac{F}{A U^2 \rho}, \frac{\sqrt{A}\, G}{U^2 \rho}, \frac{L}{\sqrt{A}}, \frac{c}{U}, \frac{\delta}{\sqrt{A}}, \frac{\mu}{\sqrt{A}\, U \rho}, \frac{p}{U^2 \rho}$$

$$dimpar_{44} := \frac{F}{A^{3/2} G}, \frac{L}{\sqrt{A}}, \frac{\mu U}{A G}, \frac{\mu c}{A G}, \frac{\delta}{\sqrt{A}}, \frac{p}{\sqrt{A}\, G}, \frac{A^{3/2} G \rho}{\mu^2}$$

$$dimpar_{46} := \frac{A}{\delta^2}, \frac{F}{G \delta^3}, \frac{L}{\delta}, \frac{c}{U}, \frac{U \mu}{G \delta^2}, \frac{p}{G \delta}, \frac{U^2 \rho}{G \delta}$$

$$dimpar_{48} := \frac{A}{L^2}, \frac{F}{G L^3}, \frac{\sqrt{\rho}\, U}{\sqrt{G}\sqrt{L}}, \frac{\sqrt{\rho}\, c}{\sqrt{G}\sqrt{L}}, \frac{\delta}{L}, \frac{\mu}{\sqrt{G}\, L^{3/2} \sqrt{\rho}}, \frac{p}{G L}$$

$$dimpar_{49} := \frac{A}{\delta^2}, \frac{F}{U \delta \mu}, \frac{\delta^2 G}{U \mu}, \frac{L}{\delta}, \frac{c}{U}, \frac{\delta p}{U \mu}, \frac{U \delta \rho}{\mu}$$

$$dimpar_{50} := \frac{A}{L^2}, \frac{F}{G L^3}, \frac{c}{U}, \frac{\delta}{L}, \frac{U \mu}{G L^2}, \frac{p}{G L}, \frac{U^2 \rho}{G L}$$

$$dimpar_{53} := \frac{A}{\delta^2}, \frac{F}{G \delta^3}, \frac{L}{\delta}, \frac{c}{U}, \frac{U \mu}{G \delta^2}, \frac{p}{G \delta}, \frac{U^2 \rho}{G \delta}$$

$$dimpar_{54} := \frac{U^2 \rho^2 A}{\mu^2}, \frac{\rho F}{\mu^2}, \frac{\mu G}{U^3 \rho^2}, \frac{L \rho U}{\mu}, \frac{c}{U}, \frac{U \delta \rho}{\mu}, \frac{p}{U^2 \rho}$$

$$dimpar_{55} := \frac{A}{\delta^2}, \frac{\delta^3 G}{F}, \frac{L}{\delta}, \frac{c}{U}, \frac{U \delta \mu}{F}, \frac{\delta^2 p}{F}, \frac{U^2 \delta^2 \rho}{F}$$

$$dimpar_{56} := \frac{A^{3/2}G}{F}, \frac{L}{\sqrt{A}}, \frac{c}{U}, \frac{\delta}{\sqrt{A}}, \frac{\sqrt{A}\,\mu U}{F}, \frac{pA}{F}, \frac{AU^2\rho}{F}$$

$$dimpar_{57} := \frac{A}{\delta^2}, \frac{F}{\delta^2 p}, \frac{G\delta}{p}, \frac{L}{\delta}, \frac{U}{c}, \frac{c\mu}{\delta p}, \frac{c^2\rho}{p}$$

$$dimpar_{58} := \frac{A}{L^2}, \frac{\rho F}{\mu^2}, \frac{L^3\rho G}{\mu^2}, \frac{L\rho U}{\mu}, \frac{L\rho c}{\mu}, \frac{\delta}{L}, \frac{L^2\rho p}{\mu^2}$$

$$dimpar_{61} := \frac{A}{\delta^2}, \frac{\delta^3 G}{F}, \frac{L}{\delta}, \frac{c}{U}, \frac{U\delta\mu}{F}, \frac{\delta^2 p}{F}, \frac{U^2\delta^2\rho}{F}$$

$$dimpar_{62} := \frac{A}{L^2}, \frac{L^3 G}{F}, \frac{LU\mu}{F}, \frac{L\mu c}{F}, \frac{\delta}{L}, \frac{L^2 p}{F}, \frac{\rho F}{\mu^2}$$

$$dimpar_{66} := \frac{U^2\rho^2 A}{\mu^2}, \frac{\rho F}{\mu^2}, \frac{\mu G}{U^3\rho^2}, \frac{L\rho U}{\mu}, \frac{c}{U}, \frac{U\delta\rho}{\mu}, \frac{p}{U^2\rho}$$

$$dimpar_{69} := \frac{A}{L^2}, \frac{F}{L^2c^2\rho}, \frac{LG}{c^2\rho}, \frac{U}{c}, \frac{\delta}{L}, \frac{\mu}{Lc\rho}, \frac{p}{c^2\rho}$$

$$dimpar_{70} := \frac{pA}{F}, \frac{\sqrt{F}\,G}{p^{3/2}}, \frac{\sqrt{p}\,L}{\sqrt{F}}, \frac{\sqrt{\rho}\,U}{\sqrt{p}}, \frac{\sqrt{\rho}\,c}{\sqrt{p}}, \frac{\sqrt{p}\,\delta}{\sqrt{F}}, \frac{\mu}{\sqrt{F}\sqrt{\rho}}$$

$$dimpar_{72} := \frac{F}{A^{3/2}G}, \frac{L}{\sqrt{A}}, \frac{\mu U}{AG}, \frac{\mu c}{AG}, \frac{\delta}{\sqrt{A}}, \frac{p}{\sqrt{A}\,G}, \frac{A^{3/2}G\rho}{\mu^2}$$

$$dimpar_{73} := \frac{F}{A^{3/2}G}, \frac{L}{\sqrt{A}}, \frac{c}{U}, \frac{\delta}{\sqrt{A}}, \frac{\mu U}{AG}, \frac{p}{\sqrt{A}\,G}, \frac{U^2\rho}{\sqrt{A}\,G}$$

$$dimpar_{74} := \frac{A}{\delta^2}, \frac{F}{G\delta^3}, \frac{L}{\delta}, \frac{U\mu}{G\delta^2}, \frac{c\mu}{G\delta^2}, \frac{p}{G\delta}, \frac{G\delta^3\rho}{\mu^2}$$

$$dimpar_{76} := \frac{F}{Ap}, \frac{\sqrt{A}\,G}{p}, \frac{L}{\sqrt{A}}, \frac{U\mu}{\sqrt{A}\,p}, \frac{\mu c}{\sqrt{A}\,p}, \frac{\delta}{\sqrt{A}}, \frac{p\rho A}{\mu^2}$$

$$dimpar_{77} := \frac{A}{\delta^2}, \frac{F}{\delta^2 p}, \frac{G\delta}{p}, \frac{L}{\delta}, \frac{\sqrt{\rho}\,U}{\sqrt{p}}, \frac{\sqrt{\rho}\,c}{\sqrt{p}}, \frac{\mu}{\delta\sqrt{p}\sqrt{\rho}}$$

$$dimpar_{78} := \frac{A}{\delta^2}, \frac{F}{c\delta\mu}, \frac{\delta^2 G}{c\mu}, \frac{L}{\delta}, \frac{U}{c}, \frac{\delta p}{c\mu}, \frac{c\delta\rho}{\mu}$$

$$dimpar_{79} := \frac{A}{\delta^2}, \frac{\delta^3 G}{F}, \frac{L}{\delta}, \frac{\delta\sqrt{\rho}\,U}{\sqrt{F}}, \frac{c\sqrt{\rho}\,\delta}{\sqrt{F}}, \frac{\mu}{\sqrt{F}\sqrt{\rho}}, \frac{\delta^2 p}{F}$$

$$dimpar_{81} := \frac{A}{\delta^2}, \frac{\delta^3 G}{F}, \frac{L}{\delta}, \frac{U}{c}, \frac{c\mu\delta}{F}, \frac{\delta^2 p}{F}, \frac{c^2\delta^2\rho}{F}$$

$$dimpar_{82} := \frac{A}{\delta^2}, \frac{F}{\delta^2 p}, \frac{G\delta}{p}, \frac{L}{\delta}, \frac{U}{c}, \frac{c\mu}{\delta p}, \frac{c^2\rho}{p}$$

$$dimpar_{83} := \frac{A}{L^2}, \frac{F}{G L^3}, \frac{U \mu}{G L^2}, \frac{\mu c}{G L^2}, \frac{\delta}{L}, \frac{p}{G L}, \frac{L^3 \rho G}{\mu^2}$$

$$dimpar_{85} := \frac{p A}{F}, \frac{\sqrt{F}\, G}{p^{3/2}}, \frac{\sqrt{p}\, L}{\sqrt{F}}, \frac{c}{U}, \frac{\sqrt{p}\, \delta}{\sqrt{F}}, \frac{U \mu}{\sqrt{F}\, \sqrt{p}}, \frac{U^2 \rho}{p}$$

$$dimpar_{86} := \frac{F}{A U^2 \rho}, \frac{\sqrt{A}\, G}{U^2 \rho}, \frac{L}{\sqrt{A}}, \frac{c}{U}, \frac{\delta}{\sqrt{A}}, \frac{\mu}{\sqrt{A}\, U \rho}, \frac{p}{U^2 \rho}$$

$$dimpar_{87} := \frac{G^2 A}{p^2}, \frac{G^2 F}{p^3}, \frac{G L}{p}, \frac{c}{U}, \frac{G \delta}{p}, \frac{G U \mu}{p^2}, \frac{U^2 \rho}{p}$$

$$dimpar_{90} := \frac{A^{3/2} G}{F}, \frac{L}{\sqrt{A}}, \frac{c}{U}, \frac{\delta}{\sqrt{A}}, \frac{\sqrt{A}\, \mu U}{F}, \frac{p A}{F}, \frac{A U^2 \rho}{F}$$

$$dimpar_{91} := \frac{F}{A^{3/2} G}, \frac{L}{\sqrt{A}}, \frac{c}{U}, \frac{\delta}{\sqrt{A}}, \frac{\mu U}{A G}, \frac{p}{\sqrt{A}\, G}, \frac{U^2 \rho}{\sqrt{A}\, G}$$

$$dimpar_{93} := \frac{A}{L^2}, \frac{F}{L c \mu}, \frac{L^2 G}{c \mu}, \frac{U}{c}, \frac{\delta}{L}, \frac{L p}{c \mu}, \frac{L \rho c}{\mu}$$

$$dimpar_{94} := \frac{A}{L^2}, \frac{F}{L U \mu}, \frac{L^2 G}{U \mu}, \frac{c}{U}, \frac{\delta}{L}, \frac{L p}{U \mu}, \frac{L \rho U}{\mu}$$

$$dimpar_{96} := \frac{G^{2/3} A}{F^{2/3}}, \frac{G^{1/3} L}{F^{1/3}}, \frac{U}{c}, \frac{G^{1/3} \delta}{F^{1/3}}, \frac{c \mu}{F^{2/3} G^{1/3}}, \frac{p}{F^{1/3} G^{2/3}}, \frac{c^2 \rho}{F^{1/3} G^{2/3}}$$

$$dimpar_{97} := \frac{G^{2/3} A}{F^{2/3}}, \frac{G^{1/3} L}{F^{1/3}}, \frac{U}{c}, \frac{G^{1/3} \delta}{F^{1/3}}, \frac{c \mu}{F^{2/3} G^{1/3}}, \frac{p}{F^{1/3} G^{2/3}}, \frac{c^2 \rho}{F^{1/3} G^{2/3}}$$

$$dimpar_{98} := \frac{G^{2/3} \rho^{2/3} A}{\mu^{4/3}}, \frac{\rho F}{\mu^2}, \frac{G^{1/3} \rho^{1/3} L}{\mu^{2/3}}, \frac{\rho^{2/3} U}{G^{1/3} \mu^{1/3}}, \frac{\rho^{2/3} c}{G^{1/3} \mu^{1/3}}, \frac{G^{1/3} \rho^{1/3} \delta}{\mu^{2/3}},$$
$$\frac{\rho^{1/3} p}{G^{2/3} \mu^{2/3}}$$

$$dimpar_{99} := \frac{p^2 A}{c^2 \mu^2}, \frac{F p}{c^2 \mu^2}, \frac{G \mu c}{p^2}, \frac{L p}{c \mu}, \frac{U}{c}, \frac{\delta p}{c \mu}, \frac{c^2 \rho}{p}$$

$$dimpar_{100} := \frac{A}{\delta^2}, \frac{F}{G \delta^3}, \frac{L}{\delta}, \frac{U \mu}{G \delta^2}, \frac{c \mu}{G \delta^2}, \frac{p}{G \delta}, \frac{G \delta^3 \rho}{\mu^2}$$

$$dimpar_{101} := \frac{A}{\delta^2}, \frac{\rho F}{\mu^2}, \frac{G \delta^3 \rho}{\mu^2}, \frac{L}{\delta}, \frac{U \delta \rho}{\mu}, \frac{c \delta \rho}{\mu}, \frac{\delta^2 \rho p}{\mu^2}$$

$$dimpar_{102} := \frac{A^{3/2} G}{F}, \frac{L}{\sqrt{A}}, \frac{U}{c}, \frac{\delta}{\sqrt{A}}, \frac{\sqrt{A}\, \mu c}{F}, \frac{p A}{F}, \frac{c^2 \rho A}{F}$$

$$dimpar_{103} := \frac{A}{L^2}, \frac{F}{G L^3}, \frac{\sqrt{\rho}\, U}{\sqrt{G}\, \sqrt{L}}, \frac{\sqrt{\rho}\, c}{\sqrt{G}\, \sqrt{L}}, \frac{\delta}{L}, \frac{\mu}{\sqrt{G}\, L^{3/2} \sqrt{\rho}}, \frac{p}{G L}$$

$$dimpar_{104} := \frac{A}{L^2}, \frac{F}{L^2 p}, \frac{G L}{p}, \frac{\mu U}{L p}, \frac{\mu c}{L p}, \frac{\delta}{L}, \frac{L^2 \rho p}{\mu^2}$$

$$dimpar_{105} := \frac{G^2 A}{p^2}, \frac{G^2 F}{p^3}, \frac{G L}{p}, \frac{U}{c}, \frac{G \delta}{p}, \frac{G \mu c}{p^2}, \frac{c^2 \rho}{p}$$

$$dimpar_{106} := \frac{A}{L^2}, \frac{F}{L^2 p}, \frac{G L}{p}, \frac{\mu U}{L p}, \frac{\mu c}{L p}, \frac{\delta}{L}, \frac{L^2 \rho p}{\mu^2}$$

$$dimpar_{108} := \frac{G^{2/3} A}{F^{2/3}}, \frac{G^{1/3} L}{F^{1/3}}, \frac{U}{c}, \frac{G^{1/3} \delta}{F^{1/3}}, \frac{c \mu}{F^{2/3} G^{1/3}}, \frac{p}{F^{1/3} G^{2/3}}, \frac{c^2 \rho}{F^{1/3} G^{2/3}}$$

$$dimpar_{109} := \frac{F}{A^{3/2} G}, \frac{L}{\sqrt{A}}, \frac{\mu U}{A G}, \frac{\mu c}{A G}, \frac{\delta}{\sqrt{A}}, \frac{p}{\sqrt{A}\, G}, \frac{A^{3/2} G \rho}{\mu^2}$$

$$dimpar_{110} := \frac{A^{3/2} G}{F}, \frac{L}{\sqrt{A}}, \frac{\sqrt{A}\sqrt{\rho}\, U}{\sqrt{F}}, \frac{\sqrt{A}\sqrt{\rho}\, c}{\sqrt{F}}, \frac{\delta}{\sqrt{A}}, \frac{\mu}{\sqrt{F}\sqrt{\rho}}, \frac{p A}{F}$$

$$dimpar_{112} := \frac{A}{L^2}, \frac{F}{G L^3}, \frac{\sqrt{\rho}\, U}{\sqrt{G}\sqrt{L}}, \frac{\sqrt{\rho}\, c}{\sqrt{G}\sqrt{L}}, \frac{\delta}{L}, \frac{\mu}{\sqrt{G}\, L^{3/2}\sqrt{\rho}}, \frac{p}{G L}$$

$$dimpar_{113} := \frac{\rho F}{\mu^2}, \frac{A^{3/2} G \rho}{\mu^2}, \frac{L}{\sqrt{A}}, \frac{\sqrt{A}\, \rho U}{\mu}, \frac{\sqrt{A}\, \rho c}{\mu}, \frac{\delta}{\sqrt{A}}, \frac{p \rho A}{\mu^2}$$

$$dimpar_{114} := \frac{G^2 A}{U^4 \rho^2}, \frac{G^2 F}{U^6 \rho^3}, \frac{G L}{U^2 \rho}, \frac{c}{U}, \frac{G \delta}{U^2 \rho}, \frac{\mu G}{U^3 \rho^2}, \frac{p}{U^2 \rho}$$

$$dimpar_{116} := \frac{A}{L^2}, \frac{F}{G L^3}, \frac{\sqrt{\rho}\, U}{\sqrt{G}\sqrt{L}}, \frac{\sqrt{\rho}\, c}{\sqrt{G}\sqrt{L}}, \frac{\delta}{L}, \frac{\mu}{\sqrt{G}\, L^{3/2}\sqrt{\rho}}, \frac{p}{G L}$$

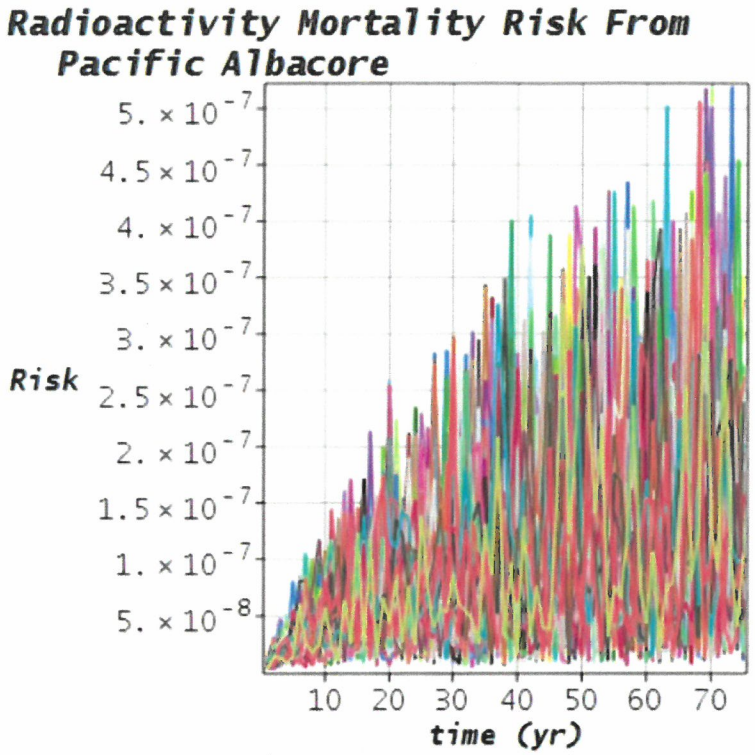
$$dimpar_{118} := \frac{A}{L^2}, \frac{F}{G L^3}, \frac{\sqrt{\rho}\, U}{\sqrt{G}\sqrt{L}}, \frac{\sqrt{\rho}\, c}{\sqrt{G}\sqrt{L}}, \frac{\delta}{L}, \frac{\mu}{\sqrt{G}\, L^{3/2}\sqrt{\rho}}, \frac{p}{G L}$$

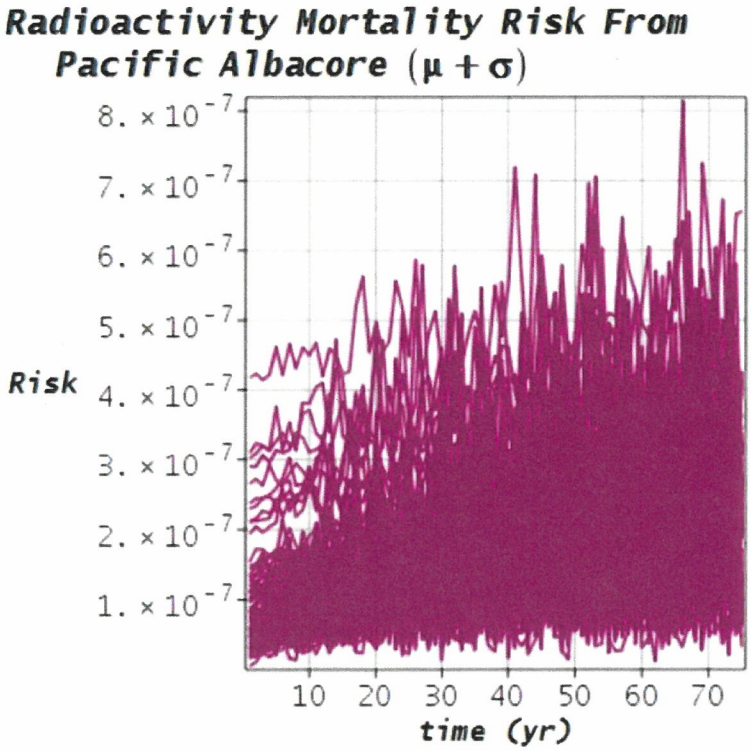
$$dimpar_{119} := \frac{G^{2/3} A}{F^{2/3}}, \frac{G^{1/3} L}{F^{1/3}}, \frac{c}{U}, \frac{G^{1/3} \delta}{F^{1/3}}, \frac{U \mu}{F^{2/3} G^{1/3}}, \frac{p}{F^{1/3} G^{2/3}}, \frac{U^2 \rho}{F^{1/3} G^{2/3}}$$

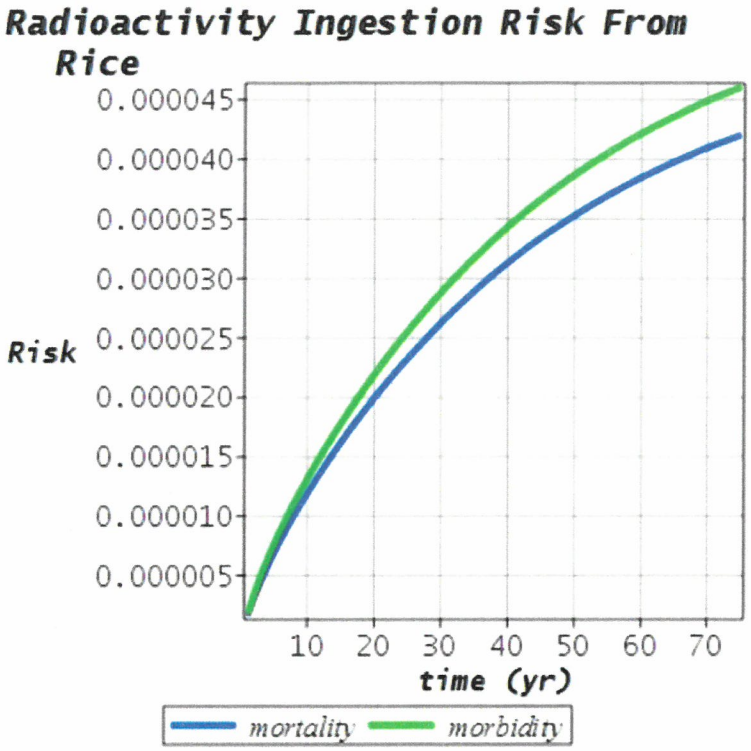
>

3.2 Radiological Accident Contaminated Tuna Off West Coast

```
restart : with(Student) : with(plots) : with(RandomTools) : with(Statistics) : wth(ListTools) :
N := 75; col1 := seq(COLOR(RGB, rand()/10^12, rand()/10^12, rand()/10^12), i = 1 ..N) : mort :=
[(9.57·10^-10, 6.88·10^-10)]; morb := [(1.39·10^-9, 1.01·10^-9)]; HLsec := [(65007360,
9.50·10^8)]; T := 75; cf := evalf(3600·24·364.2); ingest := 24.1; cic := evalf(3.7·10^10
·10^-12); c1 := evalf(200·cic); c2 := evalf(200·cic); contam := [(c1, c2)] : dos134f :=
t→ingest·contam[1]· mort[1]·e^(-ln(2)/HLsec[1]·cf·t); dos137f := t→ingest·contam[2]
· mort[2]·e^(-ln(2)/HLsec[2]·cf·t); dos134b := t→ingest·contam[1]· morb[1]
·e^(-ln(2)/HLsec[1]·cf·t); dos137b := t→ingest·contam[2]· morb[2]·exp(-ln(2)/HLsec[2]·cf·t);
xx := [seq(i, i = 1 ..T)]; yy0 := [seq(evalf(ingest·contam[1]· mort[1]
·(1 - exp(-ln(2)/HLsec[1]·cf·i))/(ln(2)/HLsec[1]·cf)), i = 1 ..T)] : yy1 :=
[seq(evalf((ingest·contam[2]· mort[2]·(1 - exp(-ln(2)/HLsec[2]·cf·i)))/(ln(2)/HLsec[2]·cf)), i = 1 ..T)] :
yy := [seq(yy1[j] + yy0[j], j = 1 ..T)]; zz0 :=
[seq(evalf((ingest·contam[1]· morb[1]·(1 - exp(-ln(2)/HLsec[1]·cf·i)))/(ln(2)/HLsec[1]·cf)), i = 1 ..T)] :
zz1 := [seq(evalf((ingest·contam[2]· morb[2]·(1 - exp(-ln(2)/HLsec[2]·cf·i)))/(ln(2)/HLsec[2]·cf)), i = 1
..T)] : zz := [seq(zz1[j] + zz0[j], j = 1 ..T)];for j from 1 to N do n := 75; un1 :=
Generate(list(float( range = 1 ..2, method = uniform), N)) : un11 :=
Generate(list(float( range = 0 ..10, method = uniform), N)) : b137 :=
Generate(list(float( range = 0.2 ..0.8, method = uniform), N)); b134 :=
Generate(list(float( range = 1 ..2, method = uniform), N)) : ingest :=
Generate(list(float( range = 1 ..3, method = uniform), N)) : dos134f := t→ingest
·b134[1]· mort[1]·e^(-ln(2)/HLsec[1]·cf·t) : dos137f := t
→ingest·b137[2]· mort[2]·(1-e^(-ln(2)/HLsec[2]·cf·t))/(ln(2)/HLsec[2]·cf) : xx2 := [seq(i, i = 1 ..T)]; yy0 :=
[seq(evalf((ingest[i]·b134[i]· mort[1]·un1[i]·(1 - exp(-ln(2)/HLsec[1]·cf·i)))/(ln(2)/HLsec[1]·cf)), i = 1
..T)] : yy1 :=
[ ( ( ingest[i]·b137[i]· mort[2]·un11[i]·(1 - exp(-ln(2)/·cf·i))) )
```

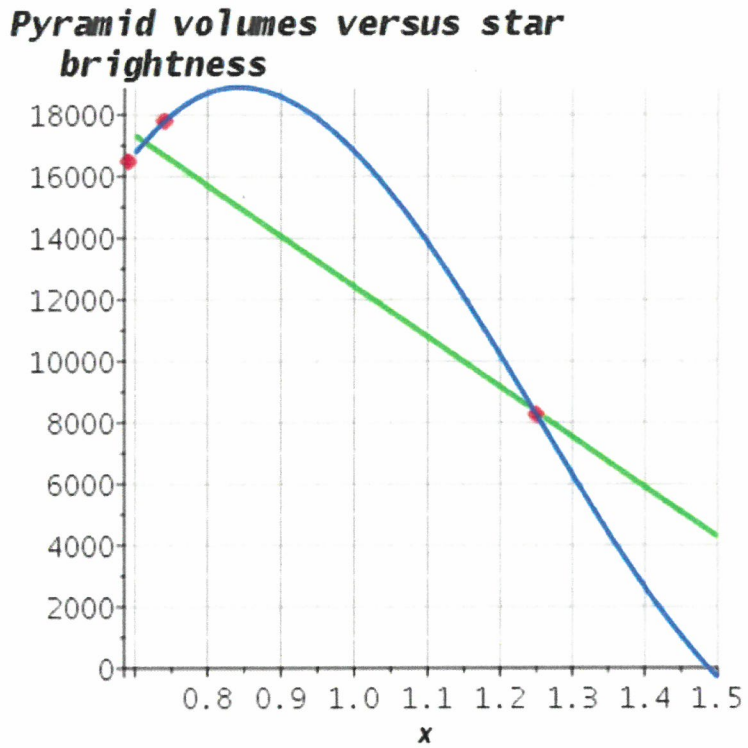

Radioactivity Mortality Risk From
Pacific Albacore
Risk
5. × 10^-7
4.5 × 10^-7
4. × 10^-7
3.5 × 10^-7
3. × 10^-7
2.5 × 10^-7
2. × 10^-7
1.5 × 10^-7
1. × 10^-7
5. × 10^-8
10 20 30 40 50 60 70
time (yr)

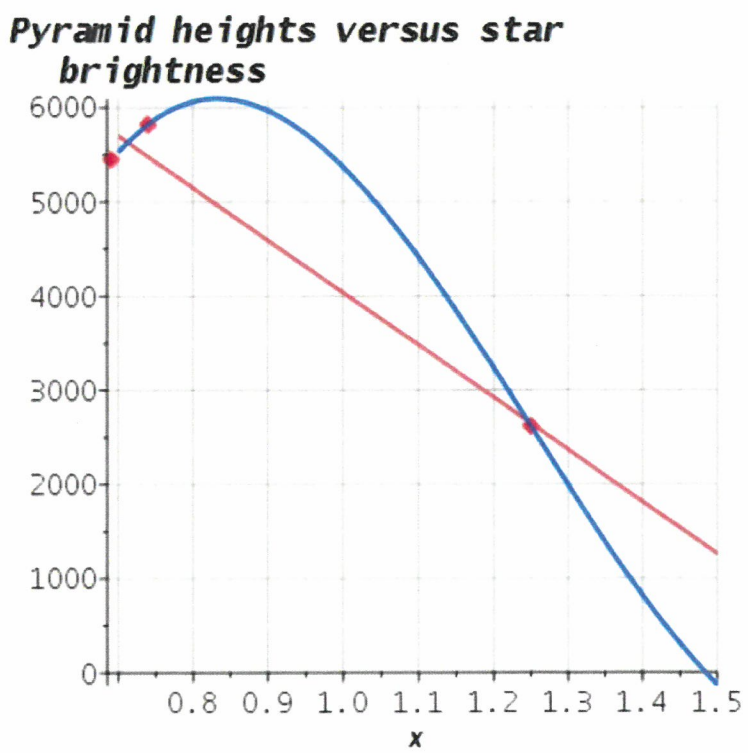

Radioactivity Mortality Risk From
Pacific Albacore (μ + σ)
Risk
8. × 10^-7
7. × 10^-7
6. × 10^-7
5. × 10^-7
4. × 10^-7
3. × 10^-7
2. × 10^-7
1. × 10^-7
10 20 30 40 50 60 70
time (yr)



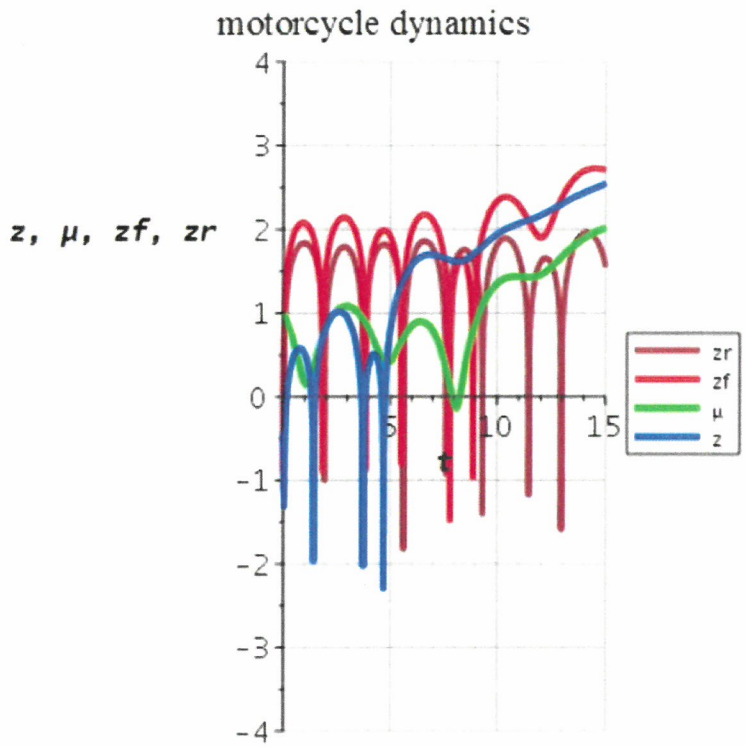
r*estart; with(Student); with(plots); mort := [1.66*10^(-9), 2.77*10^(-9)]; morb := [1.89*10^(-9), 3.03*10^(-9)]; HLsec := [65007360, 9.50*10^8]; T := 75; cf := evalf(364.2*(3600*24)); ingest := 54.6375; cic := evalf(3.7*10^10*10^(-12)); c1 := evalf(200*cic); c2 := evalf(200*cic); contam := [c1, c2]; dos134f := proc (t) options operator, arrow; ingest*contam[1]*mort[1]*e^(-ln(2)*cf*t/HLsec[1]) end proc; dos137f := proc (t) options operator, arrow; ingest*contam[2]*mort[2]*e^(-ln(2)*cf*t/HLsec[2]) end proc; dos134b := proc (t) options operator, arrow; ingest*contam[1]*morb[1]*e^(-ln(2)*cf*t/HLsec[1]) end proc; dos137b := proc (t) options operator, arrow; ingest*contam[2]*morb[2]*exp(-ln(2)*cf*t/HLsec[2]) end proc; xx := [seq(I, I = 1 .. T)]; yy0 := [seq(evalf(ingest*contam[1]*mort[1]*(1-exp(-ln(2)*cf*i/HLsec[1]))*HLsec[1]/(ln(2)*cf)), I = 1 .. T)]; yy1 := [seq(evalf(ingest*contam[2]*mort[2]*(1-exp(-ln(2)*cf*i/HLsec[2]))*HLsec[2]/(ln(2)*cf)), I = 1 .. T)]; yy := [seq(yy1[j]+yy0[j], j = 1 .. T)]; zz0 := [seq(evalf(ingest*contam[1]*morb[1]*(1-exp(-ln(2)*cf*i/HLsec[1]))*HLsec[1]/(ln(2)*cf)), I = 1 .. T)]; zz1 := [seq(evalf(ingest*contam[2]*morb[2]*(1-exp(-ln(2)*cf*i/HLsec[2]))*HLsec[2]/(ln(2)*cf)), I = 1 .. T)]; zz := [seq(zz1[j]+zz0[j], j = 1 .. T)]; pl1 := pointplot({seq([xx[i], yy[i]], I = 1 .. T)}, axes = boxed, color = blue, style = line, thickness = 3, title = 'Radioactivity*Ingestion*Risk*From*Rice', titlefont = [COURIER, Bold, 14], labels = ["time (yr)", "Risk"], labelfont = [COURIER, BOLDITALIC, 12], axesfont = [COURIER, 12], legend = mortality, gridlines, tickmarks = [10, 10]); pl2 := pointplot({seq([xx[i], zz[i]], I = 1 .. T)}, axes = boxed, color = green, style = line, thickness = 3, title = `Radioactivity Ingestion Risk From Rice  `, titlefont = [COURIER, Bold, 14], labels = ["time (yr)", "Risk"], labelfont = [COURIER, BOLDITALIC, 12], axesfont = [COURIER, 12], legend = morbidity, gridlines, tickmarks = [10, 10]); plots[display]([pl1, pl2])

3.3 Egypt's Great Pyramid's Heights versus Orion Belt Star's Brightness





3.5 Four Degrees of Freedom Motorcycle Dynamics Model

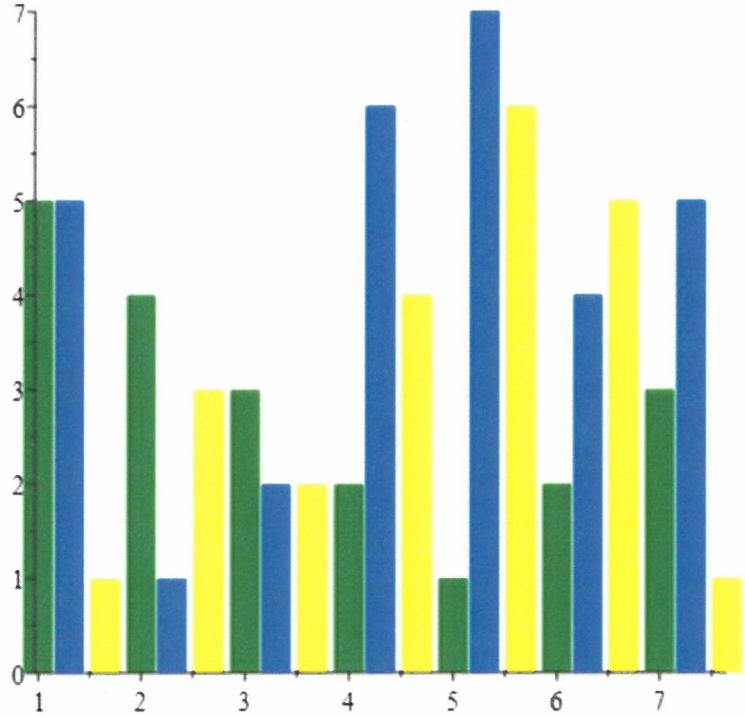



```
restart; with(Student); with(plots); with(LinearAlgebra); with(DETools); with(diffalg); with(inttrans); mf
:= 15.; mr := 18.; m := 200.; p := 1.414; b := .48; iyg := 38.; kf := 15.; kr := 24.; kp := 180; M1 := Matrix(4, 4,
[[m, 0, 0, 0], [0, iyg, 0, 0], [0, 0, mf, 0], [0, 0, 0, mr]]); M2 := Matrix(4, 4, [[kf+kr, (p-b)*kf-b*kr, -kf, -kr],
[(p-b)*kf-b*kr, (p-b)^2*kf+b^2*kf, -(p-b)*kf, b*kr], [-kf, -(p-b)*kf, b*kr], [-kr, b*kr, 0, kr+kr]]); q :=
Vector(4, [z(t), mu(t), zf(t), zr(t)]); A0 := 10.; omega1 := 1.5; tau := 100.; zer1 := Vector(4, [A0*exp(-
t/tau)*(1-t/tau)*Heaviside(t), 0, 0, 0]); eq0 := M1 . (diff(q, t, t))+M2 . q-zer1; de1 := eq0[1]; de2 := eq0[2];
de3 := eq0[3]; de4 := eq0[4]; ic1 := A0; ic2 := 0.5e-1; ic3 := 0.1e-1; ic4 := 0; z(0) := ic1; mu(0) := ic2; zf(0) :=
ic3; zr(0) := ic4; (D(z))(0) := 0; (D(mu))(0) := 0; (D(zf))(0) := 0; (D(zr))(0) := 0; delp1 := laplace(de1, t, s);
delp1s := collect(delp1, [laplace(z(t), t, s), laplace(mu(t), t, s), laplace(zf(t), t, s), laplace(zr(t), t, s)],
recursive); delp2 := laplace(de2, t, s); delp2s := collect(delp2, [laplace(z(t), t, s), laplace(mu(t), t, s),
laplace(zf(t), t, s), laplace(zr(t), t, s)], recursive); delp3 := laplace(de3, t, s); delp3s := collect(delp3,
[laplace(z(t), t, s), laplace(mu(t), t, s), laplace(zf(t), t, s), laplace(zr(t), t, s)], recursive); delp4 :=
laplace(de4, t, s); delp4s := collect(delp4, [laplace(z(t), t, s), laplace(mu(t), t, s), laplace(zf(t), t, s),
laplace(zr(t), t, s)], recursive); sol1m := solve({delp1s, delp2s, delp3s, delp4s}, {laplace(mu(t), t, s),
laplace(z(t), t, s), laplace(zf(t), t, s), laplace(zr(t), t, s)}); xt := invlaplace(rhs(sol1m[1]), s, t); yt :=
invlaplace(rhs(sol1m[2]), s, t); zt := invlaplace(rhs(sol1m[3]), s, t); wt := invlaplace(rhs(sol1m[4]), s, t); tm
:= 15; y1 := 4.; pl1 := plot(evalf(log10(abs(xt))), t = 0 .. tm, y = -y1 .. y1, axes = normal, gridlines, thickness
= 3, color = blue, style = line, title = " motorcycle dynamics ", titlefont = [COURIER, Bold, 14], labels = ["t",
"z, &mu;, zf, zr"], labelfont = [COURIER, BOLDITALIC, 12], axesfont = [COURIER, 12], legend = ["z"],
titlefont = ["ARIAL", 15], legendstyle = [font = ["HELVETICA", 9], location = right]); pl2 :=
```

```
plot(evalf(log10(abs(yt))), t = 0 .. tm, y = -y1 .. y1, axes = normal, gridlines, thickness = 3, color = green, style = line, title = "motorcycle dynamics", titlefont = [COURIER, Bold, 14], labels = ["t", "z, &mu;, zf, zr"], labelfont = [COURIER, BOLDITALIC, 12], legend = ["&mu;"], titlefont = ["ARIAL", 15], legendstyle = [font = ["HELVETICA", 9], location = right]); pl3 := plot(evalf(log10(abs(zt))), t = 0 .. tm, y = -y1 .. y1, axes = normal, gridlines, thickness = 3, color = red, style = line, title = " ", titlefont = [COURIER, Bold, 14], labels = ["t", "z, &mu;, zf, zr"], labelfont = [COURIER, BOLDITALIC, 12], legend = ["zf"], titlefont = ["ARIAL", 15], legendstyle = [font = ["HELVETICA", 9], location = right]); pl4 := plot(evalf(log10(abs(wt))), t = 0 .. tm, y = -y1 .. y1, axes = normal, gridlines, thickness = 3, color = brown, style = line, title = "motorcycle dynamics", titlefont = [COURIER, Bold, 14], labels = ["t", "z, &mu;, zf, zr"], labelfont = [COURIER, BOLDITALIC, 12], legend = ["zr"], titlefont = ["ARIAL", 15], legendstyle = [font = ["HELVETICA", 9], location = right]); plots[display]({pl1, pl2, pl3, pl4});
```

3.6 Interest Compounding, Present Value, Capital Recovery Factor calculations

3.7 Bar Chart




```
> restart : with(Student) : with(plots) : with(ColorTools) : with(plottools) :
  n := 7; x1 := seq(1.0·i, i = 1 ..n); wd := 0.3; x2 := seq(i + wd, i = 1 ..n);
  we := 0.65; x3 := seq(i + we, i = 1 ..n) ; y1 := [5, 4, 3, 2, 1, 2, 3]; y2 :=
  [5, 1, 2, 6, 7, 4, 5]; y3 := [1, 3, 2, 4, 6, 5, 1];for i from 1 to n
  do plx[i] := display(line([x1[i], 0], [x1[i], y1[i]]), linestyle = solid,
  color = "Green", thickness = 15); plx3[i] := display(line([x3[i], 0],
  [x3[i], y3[i]]), linestyle = solid, color = "yellow", thickness = 15);
  plx2[i] := display(line([x2[i], 0], [x2[i], y2[i]]), linestyle = solid,
  color = "blue", thickness = 15);end do: plty := plots[display]({seq(plx[i],
  i = 1 ..n), seq(plx2[i], i = 1 ..n), seq(plx3[i], i = 1 ..n)});
```

$$n := 7$$

$$x1 := 1.0, 2.0, 3.0, 4.0, 5.0, 6.0, 7.0$$

$$wd := 0.3$$

$$x2 := 1.3, 2.3, 3.3, 4.3, 5.3, 6.3, 7.3$$

$$we := 0.65$$

$$x3 := 1.65, 2.65, 3.65, 4.65, 5.65, 6.65, 7.65$$

$$y1 := [5, 4, 3, 2, 1, 2, 3]$$

$$y2 := [5, 1, 2, 6, 7, 4, 5]$$

$$y3 := [1, 3, 2, 4, 6, 5, 1]$$

3.8 Eigenvalues, Eigenvector Diagonalization

```
restart; with(Student); with(plots); with(LinearAlgebra); with(inttrans); with(DETools); A := Matrix(3, 3, [[0, 0, 3], [-1, -2, 2], [2, 0, 2]]); Determinant(A); Id3 := IdentityMatrix(3); x1 := Vector(3, [x, y, z]); lam := Matrix(3, 3, [[lambda, 0, 0], [0, lambda, 0], [0, 0, lambda]]); Alam := A-lam . Id3; eq := Determinant(Alam); sol := solve(eq = 0, lambda); sol1 := evalf(sol[1]); sol2 := evalf(sol[2]); sol3 := evalf(sol[3]); pts := [sol1, sol2, sol3]; complexplot(pts, style = point, symbol = solidbox, symbolsize = 16, color = red, axes = normal, gridlines); Eigenvalues(A); eig := evalf(%); eigv := evalf(Eigenvectors(A)); Av := Matrix(3, 3, eigv[2]); dtm := Determinant(Av); dtmr := evalf(1/dtm); Vinv := Adjoint(Av) . dtmr; Vinv . A . eigv[2];
```

$$A := \begin{bmatrix} 0 & 0 & 3 \\ -1 & -2 & 2 \\ 2 & 0 & 2 \end{bmatrix}$$

$$12$$

$$Id3 := \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$x1 := \begin{bmatrix} x \\ y \\ z \end{bmatrix}$$

$$lam := \begin{bmatrix} \lambda & 0 & 0 \\ 0 & \lambda & 0 \\ 0 & 0 & \lambda \end{bmatrix}$$

$$Alam := \begin{bmatrix} -\lambda & 0 & 3 \\ -1 & -2-\lambda & 2 \\ 2 & 0 & 2-\lambda \end{bmatrix}$$

$$eq := -\lambda^3 + 10\lambda + 12$$

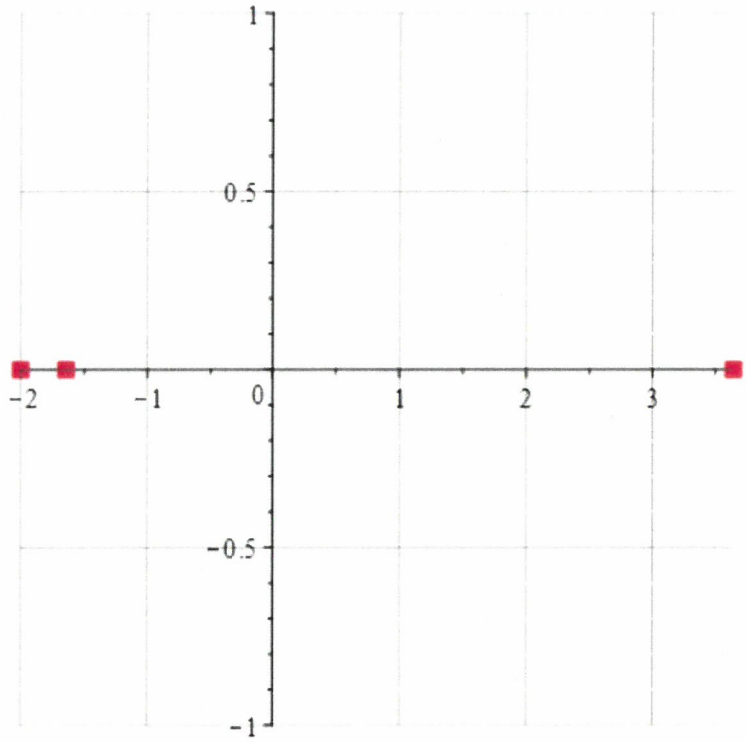
$$sol := -2, 1-\sqrt{7}, 1+\sqrt{7}$$

$$sol1 := -2.$$

$$sol2 := -1.645751311$$

$$sol3 := 3.645751311$$

$$pts := [-2., -1.645751311, 3.645751311]$$



$$eig := \begin{bmatrix} -2. \\ -1.645751311 \\ 3.645751311 \end{bmatrix}$$

$$eigv := \begin{bmatrix} 3.645751311 \\ -1.645751311 \\ -2. \end{bmatrix}, \begin{bmatrix} 0.8228756556 & -1.822875656 & 0. \\ 0.2084973779 & 10.79150262 & 1. \\ 1. & 1. & 0. \end{bmatrix}$$

$$Av := \begin{bmatrix} 0.8228756556 & -1.822875656 & 0. \\ 0.2084973779 & 10.79150262 & 1. \\ 1. & 1. & 0. \end{bmatrix}$$

$$dtm := -2.645751312$$

$$dtmr := -0.3779644729$$

$$Vinv := \begin{bmatrix} 0.377964472900000 & -0. & 0.688982236482282 \\ -0.377964472900000 & -0. & 0.311017763431096 \\ 3.99999999723454 & 1.00000000006456 & -3.49999999871411 \end{bmatrix}$$

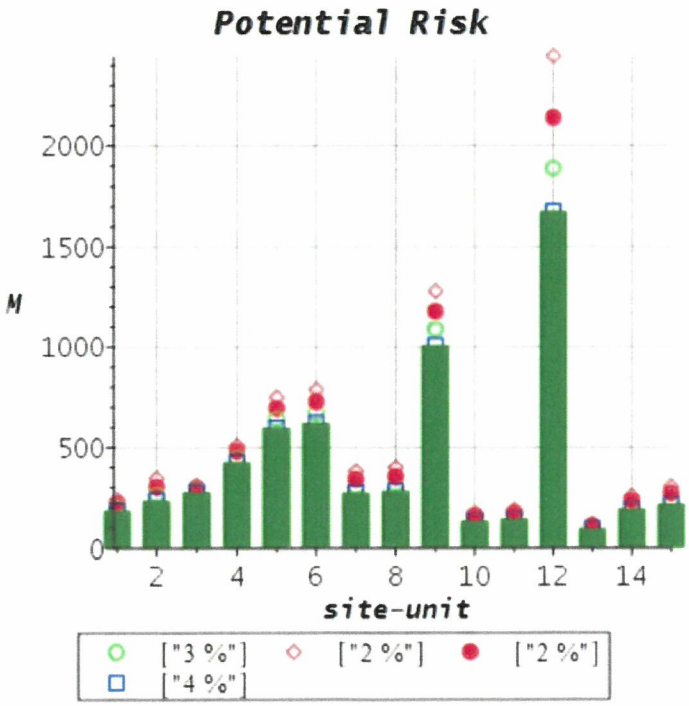
$$\begin{bmatrix} 3.64575131074879 & -9.35409083524519\,10^{-10} & 0. \\ 1.35409683466037\,10^{-10} & -1.64575131092203 & 0. \\ -4.15929068964260\,10^{-9} & -3.55928886364154\,10^{-9} & -2.00000000012913 \end{bmatrix}$$

>

3.9 Regional Nuclear Generating Risk Allocations

Study Region

Site/Unit	z pcm	Years	prem
Beaver Valley 1	-0.53	19.00	1.19
Beaver Valley 2	-0.53	30.00	1.19
Limerick 1	-0.49	7.00	9.16
Limerick 2	-0.49	12.00	9.16
Peach Bottom 2	-0.47	16.00	9.16
Peach Bottom 3	-0.49	17.00	9.16
Susquehanna 1	-0.60	25.00	1.67
Susquehanna 2	-0.61	27.00	1.63
Three Mile Island 1	-0.59	17.00	32.61
Calvert Cliffs 1	-0.52	17.00	0.41
Calvert Cliffs 2	-0.52	19.00	0.41
Hope Creek	0.96	29.00	28.80
Oyster Creek	-0.60	12.00	0.49
Salem 1	0.56	19.00	1.21
Salem 2	0.56	23.00	1.21



```
restart; with(Student); with(Finance); with(plots); with(ColorTools); ir := [0.2e-1, 0.3e-1, 0.4e-1, 0.5e-1];
cmp := 12; `&delta;oca` := 3300.; `&delta;occ` := 2.*10^4; Ccd := 1.5*10^9; Crp := evalf(1.2*10^8); Lvalue
:= 5000.; sec5 := evalf(16/3); epsilon := evalf(0.322e-4/0.641e-5); Onst := 3.*10^9; CV1 := 2.3; yrsr :=
Vector(15, Vector[column](15, [19, 30, 7, 12, 16, 17, 25, 27, 17, 17, 19, 29, 12, 19, 23])); repl :=
Vector(15, Vector[column](15, [10, 10, 7, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10])); pwr1 :=
Vector(15, Vector[column](15, [967., 967., 1170., 1170., 1320., 1320., 1320., 1320., 856., 912., 912.,
1280., 643., 1170., 1130.])); PL := [seq(evalf((1/910)*pwr1[i]), i = 1 .. 15)]; offprem := Vector(15,
Vector[column](15, [1.191, 1.191, 9.16, 9.16, 9.16, 9.16, 1.67, 1.63, 32.61, .409, .409, 28.8, .486, 1.214,
1.214])); pcm := Vector(15, Vector[column](15, [0.3980000000e-5, 0.3980000000e-5, 0.5030000000e-5,
0.5030000000e-5, 0.5420000000e-5, 0.5030000000e-5, 0.1860000000e-5, 0.1830000000e-5,
0.2370000000e-5, 0.4080000000e-5, 0.4080000000e-5, 0.4440000000e-4, 0.2050000000e-5,
0.3350000000e-4, 0.3350000000e-4])); premgeo := Vector(15, Vector[column](15, [45.3, 45.3, 143.,
143., 164., 164., 57.9, 57.2, 267., 35., 35., 322., 32.1, 42.6, 42.6])); for j to 4 do crf0[j] :=
[seq(evalf(ir[j]*(1.+ir[j])^yrsr[i]/((1.+ir[j])^yrsr[i]-1.)), i = 1 .. 15)]; Ct[j] := [seq(evalf(1./crf0[j][i]), i = 1 ..
15)]; Ctc[j] := [seq(evalf((1-exp(-ir[j]*yrsr[i]))/ir[j]), i = 1 .. 15)]; Sm[j] := [seq(evalf((1-exp(-
ir[j]*repl[i]))/(repl[i]*ir[j])), i = 1 .. 15)] end do; for j to 4 do Offex[j] := [seq(evalf((Lvalue .
premgeo[i])*Ct[j][i]*sec5), i = 1 .. 15)]; OnLT[j] := [seq(evalf((pcm[i] . `&delta;occ` . Lvalue)*Ct[j][i] .
Sm[j][i]), i = 1 .. 15)]; Onst[j] := [seq(pcm[i] . Lvalue . `&delta;oca` . Ct[j][i], i = 1 .. 15)]; Ontot[j] :=
[seq(Onst[j][i]+Offex[j][i]+OnLT[j][i], i = 1 .. 15)]; Ocd[j] := [seq(evalf((pcm[i] . Crp)*Ct[j][i]^2 . PL[i]), i = 1
.. 15)]; Rp[j] := [seq(evalf(pcm[i]*Crp*Ct[j][i]^2*PL[i]), i = 1 .. 15)]; TotInt[j] :=
[seq(evalf(Ontot[j][i]+Ocd[j][i]+Rp[j][i]), i = 1 .. 15)] end do; for j to 4 do Totext[j] :=
[seq(evalf(epsilon*TotInt[j][i]), i = 1 .. 15)]; Total[j] := [seq(evalf((CV1*Totext[j][i]+TotInt[j][i])*10^(-6)), i
= 1 .. 15)] end do;

with(plottools); "use Enter twice"; pl4 := seq(pointplot([pts[4][i]], symbol = box, color = blue, symbolsize
= 16, axes = normal, gridlines, title = 'Potential*Risk', titlefont = [COURIER, Bold, 14], labels = ["site-unit",
"M"], labelfont = [COURIER, BOLDITALIC, 12], axesfont = [COURIER, 12], legend = ["4 %"]), i = 1 .. 15);
plots[display]({pl4}); pl3 := seq(pointplot([pts[3][i]], symbol = circle, color = green, symbolsize = 16, axes
= normal, gridlines, title = 'Potential*Risk', titlefont = [COURIER, Bold, 14], labels = ["site-unit", "M"],
labelfont = [COURIER, BOLDITALIC, 12], axesfont = [COURIER, 12], legend = ["3 %"]), i = 1 .. 15);
plots[display]({pl3}); pl2 := seq(pointplot([pts[2][i]], symbol = solidcircle, color = red, symbolsize = 16,
axes = normal, gridlines, title = 'Potential*Risk', titlefont = [COURIER, Bold, 14], labels = ["site-unit",
"M"], labelfont = [COURIER, BOLDITALIC, 12], axesfont = [COURIER, 12], legend = ["2 %"]), i = 1 .. 15);
plots[display]({pl2}); pl1 := seq(pointplot([pts[1][i]], symbol = diamond, color = orange, symbolsize = 16,
axes = normal, gridlines, title = 'Potential*Risk', titlefont = [COURIER, Bold, 14], labels = ["site-unit",
"M"], labelfont = [COURIER, BOLDITALIC, 12], axesfont = [COURIER, 12], legend = ["2 %"]), i = 1 .. 15);
plots[display]({pl1}); n := 15; wd := 1.; x1 := seq(wd*i, i = 1 .. n); for i to n do plx[i] := display(line([x1[i], 0],
[x1[i], Total[4][i]]), linestyle = solid, color = "Green", thickness = 15, axes = normal, gridlines, title =
'Potential*Risk', titlefont = [COURIER, Bold, 14], labels = ["site-unit", "M"], labelfont = [COURIER,
BOLDITALIC, 12], axesfont = [COURIER, 12]) end do; pltyy := plots[display]({seq(plx[i], i = 1 .. n)}); pts[1]
:= seq([i, Total[1][i]], i = 1 .. 15); pts[2] := seq([i, Total[2][i]], i = 1 .. 15); pts[3] := seq([i, Total[3][i]], i = 1 ..
15); pts[4] := seq([i, Total[4][i]], i = 1 .. 15); plots[display]({pl1, pl2, pl3, pl4, pltyy}); nmes := Vector(15,
```

["1=Beaver Valley 1", "2=Beaver Valley 2", "3=Limerick 1", "4=Limerick 2", "5=Peach Bottom 2", "6=Peach Bottom 3", "7=Susquehanna 1", "8=Susquehanna 2", "9=Three Mile Island 1", "10=Calvert Cliffs 1", "11=Calvert Cliffs 2", "12=Hope Creek", "13=Oyster Creek", "14=Salem 1", "15=Salem 2"])

Regional Optimization

Allocating resources to a subset of the region's power plants by choosing variable, $N \leq 15$, of the sites-units while maximizing the potential risk reduction obtains, for example, using MAPLE software.

N	Sites/Units
2	7, 8
4	7, 8, 9, 13
6	1, 2, 7, 8, 9, 13
8	1, 2, 7, 8, 9, 10, 11, 13

The weights on the binary decision variable to select a site/unit are the reciprocals of the IPE probabilities of core-melt (x 10^6), s well as, offsite person-rem . Prevention of core melt is 75% and offsite risk 25%.

```
restart; with(Student); with(Optimization); offprem0 := Vector(15, Vector[column](15, [1.191, 1.191, 9.16, 9.16, 9.16, 9.16, 1.67, 1.63, 32.61, .409, .409, 28.8, .486, 1.214, 1.214])); pcm0 := Vector(15, Vector[column](15, [0.3980000000e-5, 0.3980000000e-5, 0.5030000000e-5, 0.5030000000e-5, 0.5420000000e-5, 0.5030000000e-5, 0.1860000000e-5, 0.1830000000e-5, 0.2370000000e-5, 0.4080000000e-5, 0.4080000000e-5, 0.4440000000e-4, 0.2050000000e-5, 0.3350000000e-4, 0.3350000000e-4])); offprem := [seq(evalf(1/offprem0[i]), i = 1 .. 15)]; pcm := [seq(evalf(10^6/pcm0[i]), i = 1 .. 15)]; N := 15; for k to N do NTot := k; prev := .75; mit := .25; sol1[k] := LPSolve(sum((mit*offprem[j]+prev . pcm[j])*x[j], j = 1 .. N), {seq(x[i] >= 0, i = 1 .. N), sum(x[i], i = 1 .. N) <= NTot}, maximize, assume = binary) end do; nme := vector(15, ["1=Beaver Valley 1", "2=Beaver Valley 2", " 3=Limerick 1", "4=Limerick 2", "5=Peach Bottom 2", "6=Peach Bottom 3", "7=Susquehanna 1", "8=Susquehanna 2", "9=Three Mile Island 1", "10=Calvert Cliffs 1", "11=Calvert Cliffs 2", "12=Hope Creek", "13=Oyster Creek", "14=Salem 1", "15=Salem 2"]);
```

4.0 Spherical Trigonometry Haversine Distance Calculation

```
restart; with(Student); with(plots); with(LinearAlgebra); with(MTM); nmes := ["PHL", "NPoint", "JRS", "Mecca"]; M := Matrix(4, 2, [[39.95258, -75.1652], [90., -75.1652], [31.7683, 35.2137], [21.42664, 39.82563]]); f := 1.; M1 := evalf(f . M); RE := 3962.; Lat := [seq(M1[i, 1], i = 1 .. 4)]; Long := [seq(M1[k, 2], k = 1 .. 4)]; for j to 4 do for i to 4 do A[i, j] := evalf(2*RE*arcsin(sin(.5*(Lat[i]-Lat[j]))^2+cos(Lat[j])*cos(Lat[i])*sin(.5*(Long[i]-Long[j]))^2)) end do end do; dist := Matrix(4, 4, [[A[1, 1], A[1, 2], A[1, 3], A[1, 4]], [A[2, 1], A[2, 2], A[2, 3], A[2, 4]], [A[3, 1], A[3, 2], A[3, 3], A[3, 4]], [A[4, 1], A[4, 2], A[4, 3], A[4, 4]]]);
```

$$nmes := [\text{"PHL"}, \text{"NPoint"}, \text{"JRS"}, \text{"Mecca"}]$$

$$M := \begin{bmatrix} 39.95258 & -75.1652 \\ 90. & -75.1652 \\ 31.7683 & 35.2137 \\ 21.42664 & 39.82563 \end{bmatrix}$$

$$f := 1.$$

$$M1 := \begin{bmatrix} 39.9525800000000 & -75.1652000000000 \\ 90. & -75.1652000000000 \\ 31.7683000000000 & 35.2137000000000 \\ 21.4266400000000 & 39.8256300000000 \end{bmatrix}$$

$$RE := 3962.$$

$$Lat := [39.9525800000000, 90., 31.7683000000000, 21.4266400000000]$$

$$Long := [-75.1652000000000, -75.1652000000000, 35.2137000000000, 39.8256300000000]$$

$$dist := \begin{bmatrix} 0. & 93.8285077516663 & 763.230523645900 & 3062.73664150311 \\ 93.8285077516663 & 0. & 1225.20748931032 & 2588.44694381529 \\ 763.230523645900 & 1225.20748931032 & 0. & 2985.50470742029 \\ 3062.73664150311 & 2588.44694381529 & 2985.50470742029 & 0. \end{bmatrix}$$

>

Hobby On !

Disclaimer

The applications and any opinions expressed within are the author's only and do not reflect the views of the U. S. Nuclear Regulatory Commission, the U. S. Environmental Protection Agency or other government agencies.