

W = emgr (f, g, s, t, w, pr, nf, ut, us, xs, um, xm, dp) ;

emgr – Empirical Gramian Framework (Version 5.5)

Mandatory Arguments

f	System Vector Field	(Handle)	$\mathbf{x} = \mathbf{f}(\mathbf{x}, \mathbf{u}, \mathbf{p}, t)$	i.e.: $\mathbf{f} = @(\mathbf{x}, \mathbf{u}, \mathbf{p}, t) \mathbf{A}*\mathbf{x} + \mathbf{B}*\mathbf{u} + \mathbf{F}*\mathbf{p}$;
g	Output Functional	(Handle)	$\mathbf{y} = \mathbf{g}(\mathbf{x}, \mathbf{u}, \mathbf{p}, t)$	i.e.: $\mathbf{g} = @(\mathbf{x}, \mathbf{u}, \mathbf{p}, t) \mathbf{C}*\mathbf{x} + \mathbf{D}*\mathbf{u}$;
		1	$\mathbf{y} = \mathbf{x}$	
s	System Dimensions	(Vector)	$\mathbf{s} = [\mathbf{M}, \mathbf{N}, \mathbf{Q}]$	(Inputs, States, Outputs)
t	Time	(Vector)	$\mathbf{t} = [\mathbf{h}, \mathbf{T}]$	(Time Step, Time Horizon)
w	Gramian Type	(Character)		
		'c'	Empirical Controllability Gramian (returns Wc)	
		'o'	Empirical Observability Gramian (returns Wo)	
		'x'	Empirical Cross Gramian (returns Wx)	
		'y'	Empirical Linear Cross Gramian (returns Wy)	
		's'	Empirical Sensitivity Gramian (returns {Wc, Ws})	
		'i'	Empirical Identifiability Gramian (returns {Wo, Wi})	
		'j'	Empirical Joint Gramian (returns {Wx, Wj})	

Optional Arguments

pr	Parameters	(Vector)	Column Vector of Parameters (Default: pr = 0)
		(Matrix)	Set of Parameter Columns ('s', 'i', 'j' requires two)
nf	Options Flags	(Vector)	Twelve Components (Default: nf = 0), see Option Flags
ut	Input Function	(Handle)	Input function $\mathbf{u} = \mathbf{ut}(\mathbf{t})$ (Default: ut = 1)
			Pseudorandom Binary
		1	Delta Impulse
		∞	Exponential Chirp
us	Steady-State Input	(Scalar)	Uniform Steady-State Input (Default: us = 0)
		(Vector)	Individual Steady-State Input (J x 1)
xs	Steady State	(Scalar)	Uniform Steady State (Default: xs = 0)
		(Vector)	Individual Steady States (N x 1)
um	Input Scales	(Scalar)	Uniform Maximum Input Scales (Default: um = 1)
		(Vector)	Individual Maximum Input Scales (J x 1)
		(Matrix)	Custom Input Scales (J x *)
xm	Steady-State Scales	(Scalar)	Uniform Maximum Steady-State Scales (Default: xm = 1)
		(Vector)	Individual Maximum Steady-State Scales (N x 1)
		(Matrix)	Custom Steady-State Scales (N x *)
dp	Dot Product	(Handle)	Handle to custom inner product $\mathbf{z} = \mathbf{dp}(\mathbf{x}, \mathbf{y})$
		[]	Default Matrix product

Option Flags

nf (1)	Trajectory Centering		
		0	None (Default)
		1	Initial State
		2	Final Steady State
		3	Arithmetic Average
		4	Root-Mean-Squared
		5	Midrange
nf (2)	Input Scale Sequence		
		0	Single (Default)
		1	Linear
		2	Geometric
		3	Logarithmic
		4	Sparse
nf (3)	State Scale Sequence		
		0	Single (Default)
		1	Linear
		2	Geometric
		3	Logarithmic
		4	Sparse
nf (4)	Input Transformation		
		0	Unit (Default)
		1	Inverse
nf (5)	State Transformation		
		0	Unit (Default)
		1	Inverse
nf (6)	Normalizing		
		0	None (Default)
		1	Jacobi
		2	Steady-State
nf (7)	Non-Symmetric Cross Gramian (Wx, Wy, Wj only)		
		0	Off (Default)
		1	Non-Sym. Cross Gramian
nf (8)	Extra Input (Wo, Wx, Ws, Wi, Wj only)		
		0	No Extra Input (Default)
		1	Parameter Perturbation Only
nf (9)	Center Param. Scales (Ws, Wi, Wj only)		
		0	No Centering (Default)
		1	Linear Mean Centering
		2	Logarithmic Mean Centering
nf (10)	Gramian variant (Ws, Wi, Wj only)		
	Ws	0	Input-State Average (Default)
	Ws	1	Input-Output Average
	Wi, Wj	0	Detailed Schur-Complement (Default)
	Wi, Wj	1	Approximate Schur-Complement
nf (11)	Partitioned cross Gramian (Wx, Wj only)		
		0	Full cross Gramian
		<N	Cross Gramian Partition Size
nf (12)	Partitioned cross Gramian (Wx, Wj only)		
		0	Full cross Gramian
		>0	Partition running index

Custom Solver

Global variable **ODE** to a handle with signature:
y = solver(f,g,t,x,u,p); Default: RK - SSP32

Minimal Usage: W = emgr(f,g,s,t,w);

About Info: V = emgr('version');

More info at: <http://gramian.de>