Options and UTP RSM code plug-ins to iSIGHT and Options Optimiser Plug-In to Model Center

Tony Scurr (A.D.Scurr@soton.ac.uk) **Computational Engineering Design Group** School of Engineering Sciences University of Southampton Highfield, Southampton SO17 1BJ

Options iSIGHT Plug-In GUI:



Run control parameters: Maximum iterations: 500 strained objective functions and use own internal Simulated their Annealing (SA) based optimiser for model training and optimisation.

Options Plug-In for ModelCenter:

- Objective: Incorporate the Options optimisation package with its own GUI into ModelCenter, so that a problem defined in MC can be optimised using one or more Options methods and the results stored in an MC Data Collector for further display/processing by MC facilities.
- Phoenix supply a Developer's Guide for Plug-Ins to ModelCenter. A plug-In can be written as a COM component (using C++, VB, etc) or Java component (with Java Swing GUI)

This article may be found at http://www.soton.ac.uk/~cedc/posters.html

Options Plug-In Model Set-up:

									£			
👍 Java OPTIONS Plug-In									x			
			Obj	ective Definition								
Objectiv	e Definition			Value								
Model.sparv.CDwing				0.0								
1			Cor	etraint Definitions					_			
Constraint			Value	istraint Demitions	<u> </u>	Lower Bound	Lipper Bo	und				
Model spary Clawing		0.21		0.15			777.0					
Would spar v.c. willing		0.21			0.10		-111.0					
			Des	sign Variables					_			
Variable	Variable		Value	Start Value		Lower Bound	Upper Bound					
Model.sparv.FlowSolver.alpha	3.5		7.0		0.0	7.0						
Model.sparv.CamberUpper.ucamber1b		1.5		4.0		-1.0	4.0					
1			Par	ameter Variables		1	1		-			
Paramete	r Variables			Value								
	Parameter Variables											
									1			
				1					-			
Run Optimiser	points	Add To	Model Opti	miser Method								

The main Options Plug-In GUI showing problem data and selection buttons to drive the optimisation through lower level GUIs.

Options Method Selection (Scheduling) GUI:

👙 Options Method Sele	ction										_	J,	>
Method Type			Metho	d Name									
Single point	e point		M	Method 1 Method 2		Method 3	B Meth	od 4	Method 5			í	
OPTIVAR 1			Single	point	_								
OPTIVAR 2			ADRA	INS	DA	VID	FLETCH	JO	PC)S			
LISER		_	SEEK		SIN	1PLX	APPROX	RANDOM					
NAg			OPTU	M1	OP	TUM2							
olimbing			E04U0	CF									
climbing		-	bit		dyr	namic hill						-	
pina isamina		_							· · · ·			_	
Optimise Mod	ie: MINIMISE 💌 Max	tters:	500	Max RSM It	ers	5 Pe	arameters	Debug	Outp	ut	1		
Method Type	Method Name	Nur	mber	Direction	Т	Max Iters	RSM ObjFtn	RSM Const	RSM Upda	ate I	 DOE metho	d	
RSM (Objective Ftn.)	RSM (EIF)		2.8		-1	100	4.3	0		0		0	
RSM (Constraints)	Kriging		2.8		-1	100	0	4.1		0		0	
Global	GA		4		-1	500	0	0		0		0	
climbing	dynamic hill		2.5		-1	500	0	0		0		0	
			0		-1	100	0	0		0		0	
			0		-1	100	0	0		0		0	
			0		-1	100	0	0		0		0	
	1							1	1			- u	1



Objectives:

- To incorporate the OPTIONS optimisation, DOE and RSM algorithms into iSIGHT as a plug-in.
- To incorporate the UTP developed (Stephen Leary) DOE and RSM algorithms into iSIGHT as a plugin.

iSIGHT Plug-ins:

The following four things need to be done to add an optimisation technique (i.e., couple a code) to iSIGHT:

- 1. Provide two C routines (one through a .tgen file) to create a new Tcl package and Tcl command for the optimisation technique
- 2. Use ESI C API calls to communicate between the optimisation technique and iSIGHT
- 3. Use the Technique Generator (Techgen) utility to automatically create support code needed to add your technique to iSIGHT
- 4. Any GUI interfaces need to be programmed (in Tcl/Tk) & incor-

Optimisation mode: Minimise Last method selected: dynamic hill climbing

Optimisation, DOE and RSM methods available for selection (45+ DOE, RSM and optimisation algorithms)

Run schedule selected:

Build Kriging model of Objective Function and main constraint. Minimise Kriging predictor on objective function using GA followed by local search.

	1
Para Los Filo Contesta - Od wieskodo Crast OFFICNIC Los	
Raw Log File Contents: - 3d_wingboody_3varOPTIONS.log	
RSM Search Parameters: Objective Function Method 4.100000 :Constraints Method: 4.100000 Update Flag: 0 Iterations: 100 Major Iterations: 1	
 Starting RSM evaluation for Major Iteration: 0 Calling OPTRSS with arguments of 3 4.10000 -1 27 31 1 0.01000 1.00000 Attempting to use 27 data points to construct response surface. Attempting to use 27 data points to construct constraint surface. *** WARNING *** - using existing SPM weights taken from OBJ_THETA and OBJ_EXP. *** WARNING *** - using existing SPM noise taken from OBJ_AMBDA. Largest SCV Residual is 0.63311206 SPM CCLF = -356.8337 after 572 iterations on 3 vars. & 27 data pts. Coeffs(1) = -2280 & 1.389 , Coeffs(2) = -2.582 & 2.000 Coeffs(3) = -5.442 & 2.000 Lambda = -6.6850036 Largest SCV CResidual is 0.67131453 Tcl_Error_Generated = 0 Returned variable value 0 = 0.010000 Returned variable value 2 = 2.000000 loop: 2 S00 Minimise GLEL GA Problem design variable values: (nvars) = 3 Variable input value 1 = 1.000000.3.000000 Variable input value 2 = 2.000000.00000000000 Variable input value 2 = 2.000000.00000000000000000000000000	
Status after 0 evaluations is :-	
Trial Vector	
Lwr Bound Vector Uppr Bound Variable (units)	
-0.1000000E-01<0.1000000E-01<0.4000000E-01 CAMBER() -3.000000 < 1.000000 < 3.000000 TWIST() 0. < 2.000000 < 7.000000 ALPHA()	
No of V. Boundary Violations = 0	
ОК	

Run Task Log: showing building of objective func-tion and constraint surfaces using Kriging, followed by a Genetic Algorithm (GA) optimisation.

UTP RSM Plug-In GUI:

🗙 rsmgui								_ 🗆	X
	D	OE/Response Su	rface I	Method	Selectio	n			
Max. Iterations:	100	Optimisation mo	ode:	Minimi	ise 🗕	Metho	od: RB	F opt_ rs m	
	Method Type		Meth	nod 1	Method	2 1	Method 3	Method 4	Δ
Polynomial			linear_	coeffs	linear_pre	edict qu	uad_coeffs	quad_predict	
Gaussian Ra			tra	ain	predic	t	opt_rsm		
Gradi			tra	ain	predic	t	opt_rsm		
			tra	ain	predic	t	opt_rsm		
Our all a			4	- 1	and all a				



ModelCenter Plug-In **Interfaces:**

- Tools-Trade Study Plug-In: a Trade Study can be any algorithm that repeatedly runs an analysis code to reach some objective (e.g. an optimization, DOE or parametric study).
- Component Plug-In: incorporate plug-in as a component in the Server Browser so that it can be included into Workflows. On the Workflow, click on component icon to bring up its GUI.
- Data Collector Plug-In: appears as a toolbar item in the Data Collector, and is run from there using data from the Data Collector. It is used for postprocessing operations, such as building a response surface model.

Some combinations of the above are valid, such as Trade Study + Component plug-in or Data Collector +

Sections:

- Optimisation, DOE and RSM methods available for selection (45+ DOE, RSM and optimisation algorithms).
- Run control parameters and buttons:
 - * Optimisation mode: Minimise
 - * Maximum iterations for next step: 500
 - * Maximum major iterations (updates) for RSM models: 5
 - * Call method selected parameter setting GUI
 - * Display Options Debug output file
 - * Display Options optimisation log file
- Run schedule selected:
 - ★ Build Kriging model (using **Expected Improvement Function**) of the Objective Function and main constraint. Minimise Kriging predictor on objective function using GA followed by local search.

Options Method Parameter Setting:

Method Type		Me	thod Nam	e						
Single point OPTIVAR 1 OPTIVAR 2		Method 1		Method 2	Method 3	Method 4	Method 5			
		Usi	er Func.	Shepard	RBF	Polynomial	Stochastic Stochastic		•	
		Us	er Func.	Shepard	RBF	Polynomial				
USER		Fib	onacci	Golden sect	Lagrange Int.	Hook & Jeev	Ro	senbrock		
NAa		DS	CG	DSCP	Powell	DFPS	Sir	nplex		
climbing		Co	mplex	2-mem evol	multi-mem					
		GA		SA	EP	ES	<u> </u>		-	
						·····			2	
Method Type	Method Name		Parameter Name		Value	Lower limit		Upper limit		
Global	GA		GA_NBIN		1	2	2	1	j	
			GA_NPOP		5	U	2 .	50	J	
			GA_PEN/	۹L	100,000,000		0	1,000,000,0.	•	
			GA_PBES	ST	0.	8	U			
			GA_PBES	ST DSS	0.	8	0			
			GA_PBES GA_PCR GA_PINV	ST OSS RT	0.	8	0			
			GA_PBES GA_PCR GA_PINV GA_PMU	3T 088 RT TNT	0.00	8 8 2 5	0 0 0 0 0			
			GA_PBES GA_PCRI GA_PINV GA_PMU GA_PRPT	3T 0SS RT TNT TNL	0.0.0	8 2 5 1	000000000000000000000000000000000000000	•		

porated into the overall control structure.

Basis for iSIGHT plugin development – the Tcl language:

- Tcl language consists of an interpreter that understands a base set of commands implemented in a C library.
- Tcl is powerful because it supplies a mechanism for extending the base command set with additional commands written in C or Tcl as dynamically loadable, shared libraries.
- This is the way iSIGHT itself is implemented.



Run control parameters: Maximum iterations: 100 Minimise Optimisation mode: Last method selected: **RBF** Optimise **RSM**

DOE and RSM methods including Polynomial Radial Basis Function (RBF) and Kriging algorithms, the last two with gradient enhanced algorithmic versions.

Run schedule selected: Train RBF to derive optimal sqsig parameter followed by building and optimisation of the RBF RSM surface.

The plug-in is implemented in the same way as the Options plug-in, i.e. using the iSIGHT Techgen utility. The plug-in contains a number of RSM methods, including Polynomial, RBF, Kriging, the latter two with additional gradient enhanced methods. The RSM methods only handle unconComponent plug-in.

Options Trade Study/ Component Plug-In:

What's involved:

- Develop a Java plug-in program according to the Trade Study and Component interface requirements and the JavaBeans specification.
- Develop a series of table driven (JTable/JTableX) Java Swing based GUI interfaces to drive Options.
- Integrate the Options package (Windows version) as a DLL through the JavaOptions (JNI based) interface developed for the Geodise project.
- Package the plug-in as a JAR file and register the plug-in with ModelCenter.

Change running parameters for GA algorithm.



Use Model Center RSMToolkit to generate a plot of the response surface using data generated by the 'Generate RSM surface points' option in the main GUI.

