OSTRICH Manual Addendum

This document details functionality that has been added to OSTRICH since the initial "official" version 1.6 release and subsequent "unofficial" version 1.8 release. Each section contains a heading that briefly describes an added feature. This is followed by a block of text that demonstrates the format required by OSTRICH within the "**ostIn.txt**" input file to activate and configure the new feature. Guidelines for interpreting the notation are given below:

(1) "name value" format – most OSTRICH variables are parsed according to a "name value" format where the name of a configuration variable is specified, followed by one or more whitespace characters, followed by the value that should be assigned to the variable.

(2) value options – most variables can be assigned any numerical or text value, depending on whether or not the variable represents an integer or decimal value or if it represents a text string, such as a file name, hos tname, or directory. In some cases variables can only be assigned values from a pre-defined list of choices (e.g. "yes" or "no"). For these types of variables, the acceptable values are provided in a slash-separated list (e.g. yes/no/maybe).

(3) default values – In most cases OSTRICH will use a default value for a given variable if no value is provided in the ostIn.txt input file. These default variable values (if defined) are enclosed with square brackets (e.g. '[' and ']'). Some default values are computed according to a formula and in such cases the formula has been written out within the accompanying square brackets.

(4) required values – If no default value for a variable is defined, then the user must specify a value for that variable in order to use a given feature. Such values are indicated by the use of angle brackets (e.g. '<' and '>').

(5) section tags – some variables are organized into sections which are enclosed between lines containing "Begin" and "End" tags. For example, SuperMUSE configuration variables are listed in a section that starts with the "BeginSuperMUSE" tag and ends with the "EndSuperMUSE" tag.

SuperMUSE support

```
SuperMUSE yes/[no]

BeginSuperMUSE
     AllocatorServer          [0101Prog]
     OstrichTaskerHostName     [0101Prog]
     TaskFile                 [SMuseTaskFile.txt]
     TempFile                 [SMuseTempFile.txt]
     SuccessFile              [SMuseSuccessFile.txt]
     ErrorFile                [SMuseErrorFile.txt]
     ScriptFile               [SMuseScriptFile.txt]
     ArgumentsFile            [SMuseArgumentsFile.txt]
     ClientDir                [Simulations]
     ServerDir                [FRAMESv2/Simulations]
     MaxJobTime               [120] (minutes)
EndSuperMUSE
```

Model Precision

```
NumDigitsOfPrecision [6]
```

Constant Tied Parameters

```
BeginTiedParameters
     <name> 0 <value>
EndTiedParameters
```

Shuffled Complex Evolution Algorithm

```
ProgramType ShuffledComplexEvolution

BeginSCEUA
     Budget                  [10000]
     LoopStagnationCriteria  [5]
     PctChangeCriteria       [0.01]
     PopConvCriteria         [0.001]
     NumComplexes            [sqrt(number of parameters)]
     NumPointsPerComplex     [2*(number of parameters)+1]
     NumPointsPerSubComplex  [(number of parameters)+1]
     NumEvolutionSteps       [2*(number of parameters)+1]
     MinNumOfComplexes       [sqrt(number of parameters)]
     UseInitialPoint         [yes]/no
EndSCEUA
```

Control of Random Seeds

```
RandomSeed <val>
```

Bisection Algorithm

```
ProgramType BiSectionAlgorithm

BeginBisectionAlg
     ConvergenceVal     [1E-6]
     MaxOuterIterations [50]
     MaxInnerIterations [20]
EndBisectionAlg
```

Sampling (Big Bang Big Crunch) Algorithm

```
ProgramType SamplingAlgorithm

BeginSamplingAlg
     MaxEvaluations  [100]
EndSamplingAlg
```

PSODESC Algorithm (PSO with Diversity Enhanced Shuffled Complexes)

```
ProgramType PSODESC

BeginPSODESC
     SwarmSize               [20]
     NumGenerations          [50]
     PolishingGenerations    [0]
     ConstrictionFactor      [1.00]
     CognitiveParam          [2.00]
     SocialParam             [2.00]
     InertiaWeight           [1.20]
     InertiaReductionRate    [0.10]/<VALUE>/linear
     InitPopulationMethod    [random]/quadtree/lhs
     ConvergenceVal          [0.001]
     NumComplexes            [5]
     ShuffleRate             [0.1]
EndPSODESC
```

PSODESC-GML Hybrid Algorithm

```
ProgramType PSODESC-GML
```

```
BeginPSODESC
      SwarmSize                [20]
      NumGenerations           [50]
      PolishingGenerations     [0]
      ConstrictionFactor       [1.00]
      CognitiveParam           [2.00]
      SocialParam              [2.00]
      InertiaWeight            [1.20]
      InertiaReductionRate     [0.10]/<VALUE>/linear
      InitPopulationMethod     [random]/quadtree/lhs
      ConvergenceVal           [0.001]
      NumComplexes             [5]
      ShuffleRate              [0.1]
EndPSODESC

BeginLevMar
      InitialLambda                [10.00]
      LambdaScaleFactor            [1.10]
      MoveLimit                    [0.10]
      AlgorithmConvergenceValue    [0.0001]
      LambdaPhiRatio               [0.30]
      LambdaRelReduction           [0.01]
      MaxLambdas                   [10.0]
      MaxIterations                [30.0]
      NumMultiStarts               [1]
EndLevMar
```

PSO-GML Hybrid Algorithm

```
ProgramType PSO-GML

BeginParticleSwarm
      SwarmSize                [20]
      NumGenerations           [50]
      ConstrictionFactor       [1.00]
      CognitiveParam           [2.00]
      SocialParam              [2.00]
      InertiaWeight            [1.20]
      InertiaReductionRate     [0.10]/<VALUE>/linear
      InitPopulationMethod     [random]/quadtree/lhs
      ConvergenceVal           [0.001]
EndParticleSwarm

BeginLevMar
      InitialLambda                [10.00]
      LambdaScaleFactor            [1.10]
      MoveLimit                    [0.10]
      AlgorithmConvergenceValue    [0.0001]
```

```
          LambdaPhiRatio                [0.30]
          LambdaRelReduction            [0.01]
          MaxLambdas                    [10.0]
          MaxIterations                 [30.0]
          NumMultiStarts                [1]
     EndLevMar
```

Multi-Start GML Algorithm

```
     ProgramType GML-MS

     BeginLevMar
          InitialLambda                 [10.00]
          LambdaScaleFactor             [1.10]
          MoveLimit                     [0.10]
          AlgorithmConvergenceValue     [0.0001]
          LambdaPhiRatio                [0.30]
          LambdaRelReduction            [0.01]
          MaxLambdas                    [10.0]
          MaxIterations                 [30.0]
          NumMultiStarts                [1]
     EndLevMar
```

DDS Algorithm

```
     ProgramType DDS

     BeginDDSAlg
          PerturbationValue  [0.20]
          MaxIterations      [100]
          UseInitialParamValues
          UseRandomParamValues
     EndDDSALG
```

(note: the last two DDS options are mutually exclusive)

GLUE Algorithm

```
     ProgramType GLUE

     BeginGLUE
          SamplesPerIter [10]
          NumBehavioral  [10]
          MaxSamples     [100]
          Threshold      [1000]
     EndGLUE
```

Rejection Sampler Algorithm

```
ProgramType RejectionSampler

BeginRejectionSampler
     SamplesPerIter [10]
     NumDesired     [10]
     BurnInSamples  [0]
     MaxSamples     [100]
     MinWSSE        <value>
     LikelihoodType [stedinger] / beven
     ShapingFactor  [0.5]
     TelescopeRate  [1.00]
EndRejectionSampler
```

Metropolis MCMC Algorithm

```
ProgramType MetropolisSampler

BeginMetropolisSampler
     SamplesPerIter [10]
     NumDesired     [10]
     BurnInSamples  [0]
     MaxSamples     [100]
     MinWSSE        <value>
     LikelihoodType [stedinger] / beven
     ShapingFactor  [0.5]
     TelescopeRate  [1.00]
EndMetrolopisSampler
```

Jacobian Calculation

```
ProgramType Jacobian

BeginInitParams
     <p1_val>  <p2_val> . . . <pn_val>
EndInitParams
```

Gradient Calculation

```
ProgramType Gradient

BeginInitParams
     <p1_val>  <p2_val> . . . <pn_val>
EndInitParams
```

Model Evaluation

```
ProgramType ModelEvaluation

BeginInitParams
     <p1,1_val>  <p2,1_val> . . . <pn,1_val>
     <p1,2_val>  <p2,2_val> . . . <pn,2_val>
     <p1,3_val>  <p2,3_val> . . . <pn,3_val>
        .
        .
        .
     <p1,m_val>  <p2,m_val> . . . <pn,m_val>
EndInitParams
```

Predictions

```
BeginPredictions
     <name_1>  <outfile_1> ; <keyword_1> <line_1> <col_1> <tok_1>
     <name_2>  <outfile_1> ; <keyword_2> <line_2> <col_2> <tok_2>
     <name_3>  <outfile_1> ; <keyword_3> <line_3> <col_3> <tok_3>
        .
        .
        .
     <name_n>  <outfile_n> ; <keyword_n> <line_n> <col_n> <tok_n>
EndPredictions
```

(note: if predictions are provided, OSTRICH will calculate confidence intervals and other statistics for them at the end of the calibration)

GA Crossover Rate

```
The crossover rate for the GA is now set dynamically:
    (GA crossover rate) = 1 - sqrt(1.0/(number of parameters))
```

PreserveBestModel <name_of_exe>

PreserveModelOutput yes/[no]

OstrichWarmStart yes/[no]

OstrichCaching yes/[no]

*** augmented output flag for observations