**Hackathon n° 3**

**Questionnaire sur un cas d’école parmi les applications de votre entité**

**Partie I - Spécifications fonctionnelles (matinée)**

1. Pour chacun des acteurs suivants, dire quels sont ses buts dans l’utilisation **actuelle** des outils de réduction de modèles **de votre entité**

|  |  |
| --- | --- |
| **Utilisateur boîte noire** | Buts (en 1 phrase) |
| Utilisateur du modèle réduit en boite noire, sans le connaître | * Échantillonner le modèle réduit * Charger/ouvrir le modèle réduit * ………. * ………. * ………. |

|  |  |
| --- | --- |
| **Utilisateur connaissant le modèle complet et/ou la physique** | Buts (en 1 phrase) |
| Utilisateurs connaissant le modèle complet et/ou la physique représentée | * Valider la représentativité du modèle * ………. * ………. * ………. * ………. |

|  |  |
| --- | --- |
| **Utilisateurs sachant établir un modèle réduit** | Buts (en 1 phrase) |
| Utilisateurs sachant élaborer le modèle réduit à partir du modèle complet | * Archiver/sérialiser le modèle pour le transmettre * ......... * ………. * ………. * ………. |

1. Pour chacun des acteurs suivants, dire quels sont ses buts (non listés précédemment) dans l’utilisation **future** des outils de la bibliothèque Mordicus **par votre entité**

|  |  |
| --- | --- |
| **Utilisateur boîte noire** | Buts (en 1 phrase) |
| Utilisateur du modèle réduit en boite noire, sans le connaître | * ......... * ......... * ………. |

|  |  |
| --- | --- |
| **Utilisateur connaissant le modèle complet et/ou la physique** | Buts (en 1 phrase) |
| Utilisateurs connaissant le modèle complet et/ou la physique représentée | * ......... * ......... * ………. |

|  |  |
| --- | --- |
| **Utilisateurs sachant établir un modèle réduit** | Buts (en 1 phrase) |
| Utilisateurs sachant élaborer le modèle réduit à partir du modèle complet | * D’autres méthodes de réduction * ………. |

1. Parmi ces buts, choisissez un « cas d’école », emblématique (ou du moins typique) de la pratique de la pratique de la réduction de modèle dans votre entité.
   1. Cocher les fonctionnalités génériques suivantes, listées lors du premier Hackathon, auxquelles votre cas serait susceptible de correspondre ou de faire appel :

*Utilisateur boîte noire*

* X Usage d’un modèle réduit en plan d’expérience
* Utilisation d’un modèle réduit avec des mesures in-situ (en laboratoire, sur site de production...)
* Disposer d’un “clone digital” basé sur un modèle réduit
* Générer un champ complet à partir d’un modèle réduit
* Permettre la visualisation des résultats du modèle réduit
* Archiver une liste de modèles réduits
* Gérer une taille mémoire prescrite pour l’élaboration d’un modèle réduit
* Couvrir avec un modèle réduit un sous-domaine d’usage du modèle haute fidélité
* X Garantir qu’un modèle réduit fonctionne de la même manière que le modèle haute fidélité sur un sous-domaine
* Couplage spatial entre un modèle réduit et un modèle haute fidélité

*Utilisateur connaissant le modèle complet et/ou la physique*

* Création d’un modèle réduit pour effectuer un plan d’expérience (sans exigence de méthode particulière de réduction, mais avec un contrôle de sa fiabilité)
* Comparer un modèle réduit 1 avec un modèle haute fidélité 2
* Optimisation de loi(s) de comportement sur structure complexe
* Faire interagir un modèle réduit et des opérations de Data Science

*Utilisateurs sachant établir un modèle réduit*

* Création d’un modèle réduit avec des informations provenant de mesures
* Déduire un modèle réduit 2 à partir d’un modèle réduit 1
* Permettre le calcul multi-échelles ou multi-physiques de systèmes représentés par des modèles réduits
* Permettre la visualisation du modèle réduit lui-même
* Construire un modèle réduit 3 en combinant deux modèles réduits 1 et 2
* X Évaluer l’intérêt de la démarche “Offline+Modèle réduit” par rapport à l’utilisation directe du modèle haute fidélité
* Construire un modèle réduit lorsque le maillage (voire la géométrie) change entre les différents snapshots
* Utilisation de tous les calculs intermédiaires existants pour effectuer la réduction
  1. Si votre cas ne s’apparente à aucune des fonctionnalités ci-dessus, proposer une formulation générique de votre cas :

………………………………………………………………………………………………………………………………………………

………………………………………………………………………………………………………………………………………………

………………………………………………………………………………………………………………………………………………

1. Décrivez votre cas d’usage en remplissant le tableau suivant (exemple en page …)

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE #** | Create a metamodel of a non-linear free-fall solid in a viscous fluid  This is an analytical toy example. | | |
| **Context of use** | The context is uncertainty propagation in a field model. Typically we want to build a reduced model (inline), then assess its validity against the real model (offline). Finally we want to propagate uncertainties through the reduced model. | | |
| **Scope** | Inline and offline components of Mordicus | | |
| **Level** | Primary task | | |
| **Primary actor** | An R&D engineer with knowledge of reduced-order modelling techniques | | |
| **Stakeholder & Interests** | **Stakeholder** | | **Interest** |
|  | User of a reduced-order model in his field | | Wants fast evaluation of the model to explore the model response to given inputs |
|  | R&D engineer with knowledge of uncertainty propagation methodology | | Wants to perform a sensitivity analysis, thredshold exceedance simulation |
| **Preconditions** | * A computer code with a custom language-specific wrapper * Or an already-computed design of experiment | | |
| **Success End Condition** | A representative reduced-order model | | |
| **Failed End Protection** | A visually non-representative reduced-order model | | |
| **Trigger** | Begin of the uncertainty propagation study | | |
| **DESCRIPTION** | **Step** | **Action** | |
|  | 1 | User provides X/Y data from the model evaluations and the mesh  The input is 4-d, the output is 1-d on a 1-d mesh (regular time grid)  The output timeseries are evaluated on the same time grid  There should not be nans or missing values | |
|  | 2 | User: demands the model to be built (inline) | |
|  | 3 | System: checks input data (X/Y size, outputs vs mesh size) | |
|  | 4 | System: apply KarunhenLoeve SVD decomposition, projection of output series, perform functional chaos in the decomposition space | |
|  | 5 | User: demands evaluation on validation input sample (offline) | |
|  | 6 | System: checks input sample | |
|  | 7 | System: evaluates the model | |
|  | 8 | ………………………………………………………………………………………………………………………………..  ……………………………………………………………………………………………………………………………….. | |
|  |  | ………………………………………………………………………………………………………………………………..  ……………………………………………………………………………………………………………………………….. | |
|  |  | ………………………………………………………………………………………………………………………………..  ……………………………………………………………………………………………………………………………….. | |
| **EXTENSIONS** | **Step** | **Branching Action** | |
|  | 3a | Size of inputs differs from the outputs size (the number of input points vs the number of output trajectories)  Inform user and clean exit | |
|  | 3b | The output discretization size differs from the provided mesh size  Inform user and clean exit | |
|  | 6a | The input sample dimension differs from the reduced-order input dimension  Inform user and clean exit | |
|  |  | ………………………………………………………………………………………………………………………………..  ……………………………………………………………………………………………………………………………….. | |
|  |  | ………………………………………………………………………………………………………………………………..  ……………………………………………………………………………………………………………………………….. | |
|  |  | ………………………………………………………………………………………………………………………………..  ……………………………………………………………………………………………………………………………….. | |
| **I/O VARIATIONS** |  | **Branching Action** | |
|  | 1 | <list of variations> | |
|  |  | …………………………………………………………………………………………………………………………………..  ………………………………………………………………………………………………………………………………….. | |
|  |  | …………………………………………………………………………………………………………………………………..  ………………………………………………………………………………………………………………………………….. | |
|  |  | …………………………………………………………………………………………………………………………………..  ………………………………………………………………………………………………………………………………….. | |
| **RELATED INFORMATION** |  | <list here related documentation to this use case, and whether or not you are ready to make it available to other members of Mordicus> | |
|  |  | …………………………………………………………………………………………………………………………………..  ………………………………………………………………………………………………………………………………….. | |
|  |  | …………………………………………………………………………………………………………………………………..  ………………………………………………………………………………………………………………………………….. | |
|  |  | …………………………………………………………………………………………………………………………………..  ………………………………………………………………………………………………………………………………….. | |
|  |  | …………………………………………………………………………………………………………………………………..  ………………………………………………………………………………………………………………………………….. | |

**Exemple (sous forme de tableau)**

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE #EDF.01** | Apply the offline stage for a linear parametric parabolic problem | | |
| **Context of use** | Expert user establishes the needed autonomous data structure for running a reduced-order simulation of a parametric parabolic problem, with operators linearly dependent of the parameters and the state. For instance, a linear transient thermal analysis with an explicit parameter variation of the conductivity, capacity or exchange coefficient falls into this case. | | |
| **Scope** | The offline component of Mordicus library | | |
| **Level** | Primary Task | | |
| **Primary actor** | User with knowledge of reduced-order modelling techniques | | |
| **Stakeholder & Interests** | **Stakeholder** | | **Interest** |
|  | User of a reduced-order model in his field | | wants a reduced-order model that behaves as the full-order one from his point of view;  wants clear and direct access to varying parameters and physical case description (for control) |
|  | Developer of a leightweight field-specific application | | wants an clear and portable output data structure with minimal software dependencies |
| **Preconditions** | Have available:   * a nominal full-order case of study of a phenomenon governed by parabolic equations * its result fields for all targeted time steps at least one parameter value, possibly in different folders but in the same filesystem * pre-computed full-order operators : C (unit “capacity” matrix), K (unit “conductivity” matrix), f (unit “source” right-hand side vector), fb (unit “Robin condition” right-hand side vector, zone-wise on the border), Kd (unit “Robin condition” right-hand side matrix, zone-wise on the border). Each right-hand side vector is accompanied by one time-evolution file | | |
| **Success End Condition** | The expected autonomous data structure was produced | | |
| **Failed End Protection** | Only the responsible process is stopped. User gets informed about missing or erroneous input data, or algorithm configuration at failure | | |
| **Trigger** | User enters the offline component of Mordicus while announcing a parabolic problem | | |
| **DESCRIPTION** | **Step** | **Action** | |
|  | 1 | User: fills user input for the offline stage  Demanded input data: parameter values (at least one) and a path to one time-history result file for each; pre-computed operators C, K as path to different files; truncation criterion; path for output data structure  Optional input data: f and fb with their time-evolution files; Kb (zones should be consistent); format for the description of the operators in the output data structure (one of ‘MED’, ‘numpy’… numpy is default); format for time-evolution files (csv…); path for log files of the offline phase (for diagnosis); capacity, conductivity or exchange coefficients if fixed | |
|  | 2 | User : demands execution of the offline phase | |
|  | 3 | System : verifies that input data is provided and valid. Extracts initial state from provided results. | |
|  | 4 | System : applies the progressive RB algo (greedy selection, appeals to incremental SVD only), produces a reduced-order basis | |
|  | 5 | System : produces compressed operators for unitary M, C, f and initial state vector f0 by algebraic projection | |
|  | 6 | System : packs the following into an output data structure: compressed operators, time-evolutions, fixed coefficients, and either links to high-fidelity file results or assumed mesh and time discretization. Writes it to a file at the location specified by the user. | |
|  | 7 | System : notifies use of completion | |
| **EXTENSIONS** | **Step** | **Branching Action** | |
|  | 3a | Submitted data is incomplete (some required input is not provided) :  ask missing data from the user | |
|  | 3b | Submitted data is erroneous due to wrong file path:  Inform user of erroneous file path, save existing input data, clean exit | |
|  | 3c | Not all provided results refer to the same mesh or zones of provided operators Kd, fd are not consistent :  Inform user and clean exit | |
|  | 3d | Time-history of the input computation results are not consistent with each other or with the provided time-evolutions :  Inform user and clean exit | |
|  | 4a | Dimensions of input results are not compatible :  Inform user of responsible files and time steps, exit | |
|  | 5a | Incompatible size error occurs on the projection:  Inform the user of the nature of the responsible matrices/vectors, exit | |
|  | 6a | Invalid or taken output file location:  Generate valid file path, save there, inform user, continue | |
| **I/O VARIATIONS** |  | **Branching Action** | |
|  | 3 | I/O variation: Initial state may be given explicitly by the user if not present in the initial results | |