CSMA 2022
15ème Colloque National en Calcul des Structures
16-20 Mai 2022, Presqu'île de Giens (Var)

New functionalities of Versions 3.3, 3.4 and 4.0 of the TFEL/MFront project and Version 1.0, 1.1 and 1.2 of the MGIS project

Thomas Helfer[(1)], Jérémy Hure [(2)], Mohamed Shokeir [(3)], Olivier Fandeur[(4,5)], Olivier Jamond[(6)], Jean-Philippe Mathieu[(7)], Simon Raude [(8)], Dominique Geoffroy[(9)], Jérémy Bleyer [(10)], Thomas Nagel [(11)], Guillaume Latu [(12)]

[(1)] *CEA, DES/IRESNE/DEC/SESC/LSC, Département d'Études des Combustibles, Cadarache, France, thomas.helfer@cea.fr*
[(2)] *CEA, DES/ISAS/DMN/SEMI/LCMI, Département des Matériaux pour le Nucléaire, Saclay, France, jeremy.hure@cea.fr*
[(3)] *CEA, DES/ISAS/DMN/SEMI/LCMI, Département des Matériaux pour le Nucléaire, Saclay, France, mohamed.shokeir@cea.fr*
[(4)] *CEA, ISAS/DES/DM2S/SEMT/LM2S, Département de Modélisation des Systèmes et des Structures, Saclay, France, olivier.fandeur@cea.fr*
[(5)] *IMSIA, UMR 8193, CNRS-EDF-CEA-ENSTA*
[(6)] *CEA, ISAS/DES/DM2S/SEMT/LM2S, Département de Modélisation des Systèmes et des Structures, Saclay, France, olivier.jamond@cea.fr*
[(7)] *EDF R&D, Département MMC, jean-philippe.mathieu@edf.fr*
[(8)] *EDF R&D, Département ERMES, simon.raude@edf.fr*
[(9)] *EDF R&D, Département ERMES, dominique.geoffroy@edf.fr*
[(10)] *Laboratoire Navier UMR 8205 (École des Ponts ParisTech - Université Gustave Eiffel - CNRS), France, jeremy.bleyer@enpc.fr*
[(11)] *TU Bergakademie Freiberg, Freiberg, Germany, thomas.nagel@ifgt.tu-freiberg.de*
[(12)] *CEA, DES/IRESNE/DEC/SESC/LSC, Département d'Études des Combustibles, Cadarache, France, guillaume.latu@cea.fr*

Abstract — `MFront` is a tool which allows easy implementation of arbitrarly complex mechanical behaviours in an efficient way. Those implementations are portable between various finite element solvers and solvers based on FFT. `MFront` is part of the open-source `TFEL` project.

The purpose of this paper is to highlight a selected set of features introduced in Versions 3.3, 3.4 and 4.0 of the `TFEL` project. The paper also describes the `MFrontGenericInterfaceSupport` project which allows to integrate `MFront` behaviours in existing open-source or commercial solvers.

Contents

Introduction **2**

1 Overview of `TFEL`, `MFront`, `MTest` and `MGIS` **2**
1.1 Overview of the TFEL libraries . . . 2
1.1.1 The `TFEL/Math` library . . . 2
1.1.2 The `TFEL/Material` library . . . 3
1.2 Overview of the `MFront` code generator . . . 3
1.3 The `generic` interface and the `MFrontGenericInterfaceSupport` project . . . 3
1.3.1 The `mgis.fenics` module . . . 4
1.4 Material knowledge management, the `MFrontGallery` project . . . 4
1.5 Distributions and installation . . . 5

2 Main improvements since Version 3.2 **5**
2.1 Support for generalized behaviours (Version 3.3) . . . 5

2.2 Unicode Support (Version 3.3) . . . 5
2.3 Extension of the StandardElastoViscoPlasticity brick to porous materials (Version 3.4) . . . 6
2.4 Major rewrite of the TFEL/Math library (Version 4.0) . . . 6
2.5 Support of quantities (Version 4.0) . . . 7
2.6 Updated documentation and tutorials . . . 8

Conclusions and future works 8

References 9

Introduction

TFEL/MFront has been actively developped since Version 3.2, which was described in the same conference in 2019 (1), by a large community of academic and industrial users as demonstrated by the various talks at the MFront user days[1] and the list of publications using MFront[2].

This paper is devoted to highlight a selected set of features introduced in Versions 3.3.x, 3.4.x and 4.0 of the TFEL/MFront project. The paper also describes the MFrontGenericInterfaceSupport (MGIS) project which allows to integrate MFront behaviours in existing open-source or commercial solvers.

The interested reader may refer to the release notes of those versions for a comprehensive and detailled description:

- https://thelfer.github.io/tfel/web/release-notes-3.3.html
- https://thelfer.github.io/tfel/web/release-notes-3.4.html
- https://thelfer.github.io/tfel/web/release-notes-4.0.html

This paper is organized a follows:

- Section 1 provides an overview of TFEL, MFront and MTest and MGIS
- Section 2 describes the main improvements to TFEL/MFront since Version 3.2.

1 Overview of TFEL, MFront, MTest and MGIS

The TFEL project is an open-source collaborative development of the French Alternative Energies and Atomic Energy Commission (CEA) and Électricité de France (EDF) in the framework of the PLEIADES plateform (2). TFEL provides mathematical libraries which are the basis of the MFront code generator and the MTest solver (3, 4).

1.1 Overview of the TFEL libraries

1.1.1 The TFEL/Math library

The TFEL/Math library provides:

- A linear algebra engine with mathematical objects (tensors of arbitrary orders) and operations on those objets required to express the constitutive equations in an efficient and natural manner, i.e. as close as possible to the mathematical expressions common in the engineering sciences. These mathematical objects can have units allowing the compiler to perform dimensional analysis at compile-time.
- A framework to build non linear solvers for small sized problems. Efficient and robust implementations of several classical non linear algorithms (Newton-Raphson, Broyden, Levenberg-Marquart, etc.) are provided.

[1] https://github.com/thelfer/tfel-doc/tree/master/MFrontUserDays
[2] https://thelfer.github.io/tfel/web/publications.html

1.1.2 The TFEL/Material library

The TFEL/Material library provides implementations of:

- Various utility functions frequently required in constitutive modelling (computation of the Lamé coefficients, computation of the Hill tensors).
- Various stress criteria and their derivatives with respect to the stress tensor (Hill 1948, Hosfort 1978, Gurson 1977, Gurson-Tvergaard-Needlman 1984, Barlat 2004, Mohr-Coulomb, etc..).

1.2 Overview of the MFront code generator

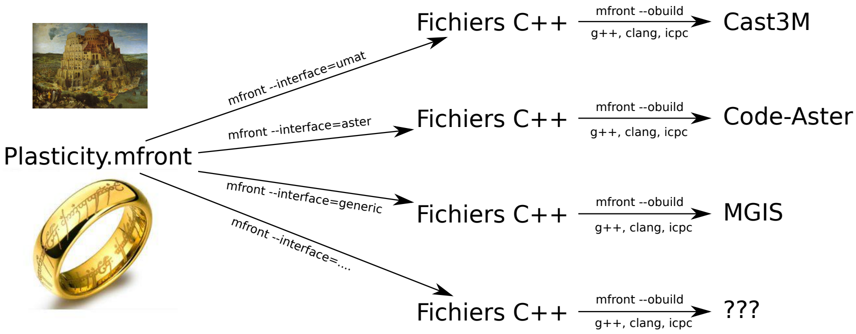

Figure 1: Principle of MFront

MFront translates a set of closely related domain specific languages into plain C++ on top of the TFEL libraries.

Those languages are meant to be easy to use and learn by researchers and engineers and cover three kinds of material knowledge: material properties (Young's modulus, thermal conductivity, etc.), mechanical behaviours[3] and simple point-wise models (such as material swelling under irradiation used in fuel performance codes). Concerning behaviours, the following kinds of behaviours are supported:

- small and finite strain mechanical behaviours.
- cohesive zone models.
- generalized behaviours as detailled in Section 2.1.

Authors of MFront paid particular attention to the robustness, reliability and numerical efficiency of the generated code, in particular for mechanical behaviours: various benchmarks show that MFront implementations are competitive with native implementations available in the Cast3M, code_aster,[4] Europlexus Abaqus/Standard Abaqus/Explicit 9) solvers and in the Cyrano3 (10) and Galileo (11) fuel performance codes.

Portability is also a very important issue: a behaviour written in MFront shall be usable in any solver for which an interface exists. In addition to the aforementioned solvers, interfaces exist for: Ansys, ZMat, CalculiX, DianaFEA.

1.3 The generic interface and the MFrontGenericInterfaceSupport project

To limit the number of interfaces supported by MFront, an interface called generic has been introduced, along with the MFrontGenericInterfaceSupport project which provides to solver developpers tools (functions, classes, bindings, etc...) to handle behaviours generated by this generic interface.

[3] Among the many projects aiming at easing the implementation of mechanical behaviours (see for example (5–7)), MFront can be compared to the ZebFront code generator which is part of ZMat library. A comprehensive comparison between these two solutions has been presented at the ZSet User Meeting (8).

[4] *salome_meca : plateforme de simulation de mécanique pour les études et la recherche.* M. Abbas et al. This conference.

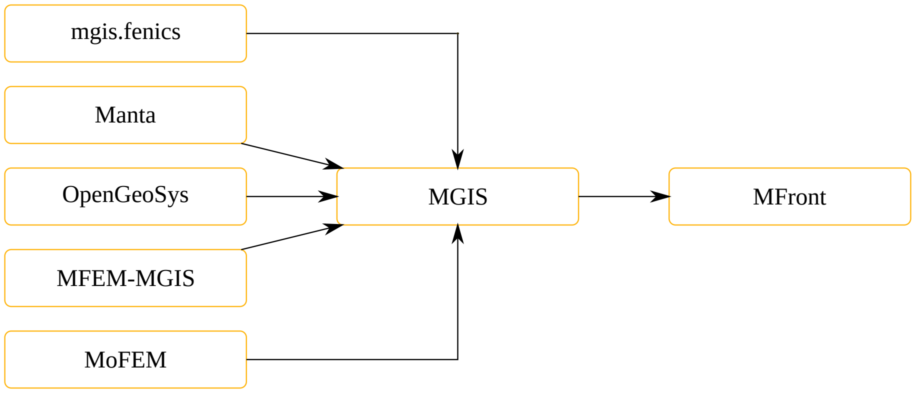

Figure 2: Principle of `MGIS`

The `MFrontGenericInterfaceSupport` project has already been integrated or tested in many solvers (12), including `mgis.fenics`, `OpenGeoSys` (13), `Manta`[5], `MFEM-MGIS`, `XPer`[6] (14), `MOOSE` `MoFEM`, `Disk++`, `Kratos Multiphysics`, `OOFEM`, `JuliaFEM`, `NSPFEM2D` (15), `esys.escript`, `DUNE`, etc.

1.3.1 The `mgis.fenics` module

The `mgis.fenics` module has been written to leverage the `FEniCS` platform for solving abitrary partial differential equations and `MFront` for describing the local description of the material constitutive equations, including generalized mulitphysics nonlinear behaviours i.e. including multiple gradients and dual flux variables. Section 2.1 provides a list of available tutorials provided with `mgis.fenics`.

1.4 Material knowledge management, the `MFrontGallery` project

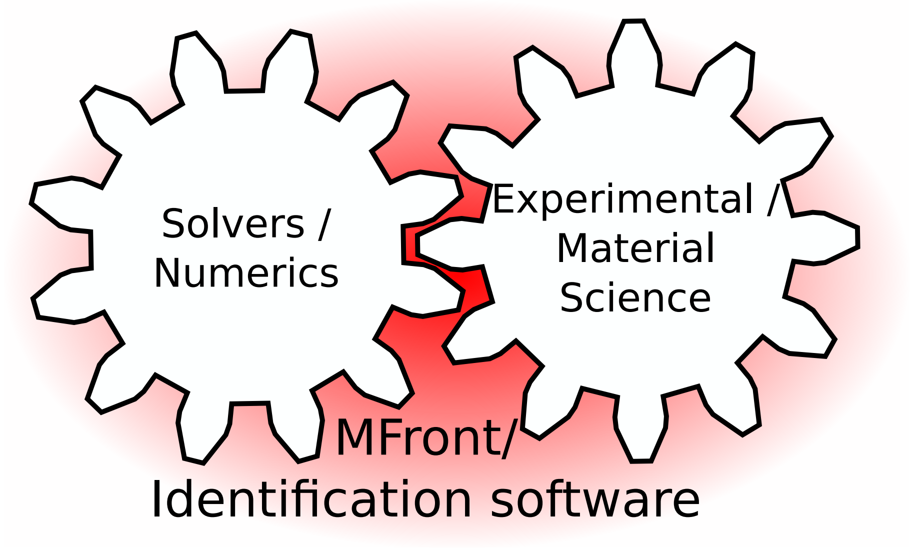

Figure 3: Separation from solvers and material knowledge thanks to `MFront`

The need to guarantee the quality of engineering studies has never been so high and is constantly growing.

This section is based on the assumpations that the knowledge about materials (material properties, behaviours) can be separated from the solvers of interest, as illustrated by Figure 3:

[5]`MANTA`: un code HPC généraliste pour la simulation de problèmes complexes en mécanique.". O. Jamond et al. This conference.

[6]*Xper: une plateforme pour la simulation numérique distribuée d'interactions multiphysiques entre corps.* F. Péralès et al. This conference.

Every part of a study must be covered by strict QA procedures

- The (finite element) solver on the one hand.
- The material knowledge (material properties, mechanical behaviours, simple point-wise models) and experimental data on the other hand.

The `MFrontGallery` project has been created to help achieve the second task. It mostly consists a collection of `cmake` functions that are meant to ease the compilation of `MFront` source files for all supported interfaces and the execution of unit tests. This project is hosted on here: https://github.com/thelfer/MFrontGallery.

1.5 Distributions and installation

`TFEL/MFront` is distributed with `Cast3M`, `code_aster` and `Salome Méca`.

Thanks to its community of users, `TFEL/MFront` is also part of various `Linux` distributions, the `FreeBSD` ports and an official package for `MINGW` exists.

For keeping up with the lastest developments, `TFEL/MFront` and `MGIS` can be built from sources, but we highly recommend using the `spack` package manager as follows:

```
$ spack install tfel@master mgis@master
```

2 Main improvements since Version 3.2

2.1 Support for generalized behaviours (Version 3.3)

Generalized behaviours relate an arbitrary number of gradients to their conjugated thermodynamic forces. In implicit solvers, generalized behaviours must also provide so called tangent operator blocks to build the stiffness matrix.

Generalised behaviours allow to implement non local behaviours, strongly coupled multi-physics behaviours, behaviours of structural elements such as beams, as illustrated by various demos from the `mgis.fenics` module:

- Stationary non-linear heat transfer
- Stationary non-linear heat transfer: 3D problem and performance comparisons
- Transient heat equation with phase change
- Monolithic transient thermoelasticity
- Multiphase model for fiber-reinforced materials
- Phase-field approach to brittle fracture

Other applications include Cosserat plasticity (16), strongly coupled thermo-mechanical models of single crystals (17), micromorphic approaches to brittle fracture[7], elasto-plasticity,[8] etc.

2.2 Unicode Support (Version 3.3)

Some recent scientific programming languages like `Julia` or ∇ allow the usage of (a subset of) unicode characters (18, 19). This allows a much more readable code, very close to the mathematical expressions.

Here is an example of the implementation of a Norton behaviour:

```
@Integrator{
  const auto σᵉ = sigmaeq(σ);
  const auto iσᵉ = 1 / (max(σᵉ, real(1.e-12) · E));
```

[7] *Une approche micromorphe de l'endommagement de matériaux quasi-fragiles: implémentation numérique et lien avec la méthode par champ de phase.* O. Fandeur et al. This conference.

[8] *Formulation thermodynamique et cadre variationnel pour des modèles d'élastoplasticité à état critique.* G. Bacquaert et al. This conference.

```
  const auto vᵖ = A ⋅ pow(σᵉ, nn);
  const auto ∂vᵖ/∂σᵉ = nn ⋅ vᵖ ⋅ iσᵉ;
  const auto n = 3 ⋅ deviator(σ) ⋅ (iσᵉ / 2);
  // Implicit system
  fεᵉˡ += Δp ⋅ n;
  fp -= vᵖ ⋅ Δt;
  // jacobian
  ∂fεᵉˡ/∂Δεᵉˡ += 2 ⋅ μ ⋅ θ ⋅ dp ⋅ iσᵉ ⋅ (Mᵉ - (n ⊗ n));
  ∂fεᵉˡ/∂Δp = n;
  ∂fp/∂Δεᵉˡ = -2 ⋅ μ ⋅ θ ⋅ ∂vᵖ/∂σᵉ ⋅ Δt ⋅ n;
} // end of @Integrator
```

The supported subset of unicode characters is fully detailed on the TFEL website.

2.3 Extension of the `StandardElastoViscoPlasticity` brick to porous materials (Version 3.4)

The `StandardElastoViscoPlasticity` brick as been introduced in Version 3.3 to implement behaviours in a declarative, clear and concise way using pre-implemented stress criteria and kinematic hardening rules.

The `StandardElastoViscoPlasticity` brick has been extended to support porous (visco-)plastic flows which are typically used to model ductile failure of metals (20, 21) by describing the evolution of the porosity. Standard models for ductile failure have been implemented (Gurson-Tvergaard-Needleman (22), Rousselier-Tanguy-Besson (23)), as well as various nucleation models (24).

By default, an original implicit algorithm is used where the porosity evolution is treated in a staggered approach (20, 21).

Complex porous plastic models can be implemented in a modular manner as follows:

```
@Brick StandardElastoViscoPlasticity{
  stress_potential : "Hooke" {young_modulus : 70e3, poisson_ratio : 0.3},
  inelastic_flow : "Plastic" {
    criterion : "GursonTvergaardNeedleman1982" {
      f_c : 0.04, f_r : 0.056, q_1 : 2., q_2 : 1., q_3 : 4.},
    isotropic_hardening : "Linear" {R0 : 274},
    isotropic_hardening : "Voce" {R0 : 0, Rinf : 85, b : 17},
    isotropic_hardening : "Voce" {R0 : 0, Rinf : 17, b : 262}
  }
  nucleation_model : "Chu_Needleman" {
    An : 0.01, pn : 0.1, sn : 0.1 },
};
```

Figure 4 outlines crack propagation simulations carried with the above presented material behavior and the `Cast3M` solver.

2.4 Major rewrite of the `TFEL/Math` library (Version 4.0)

Version 4.0 of the `TFEL` project is based on the `C++-17` standard. The new features of the `C++-17` standard allowed for a major rewrite of the `TFEL/Math` library, the implication of which is fully described in the release notes.

The main goal of this rewrite is to reduce the code size, improve the overall maintainability and prepare future evolutions such as the port to GPUs.

From the user point of view, this rewrite introduced:

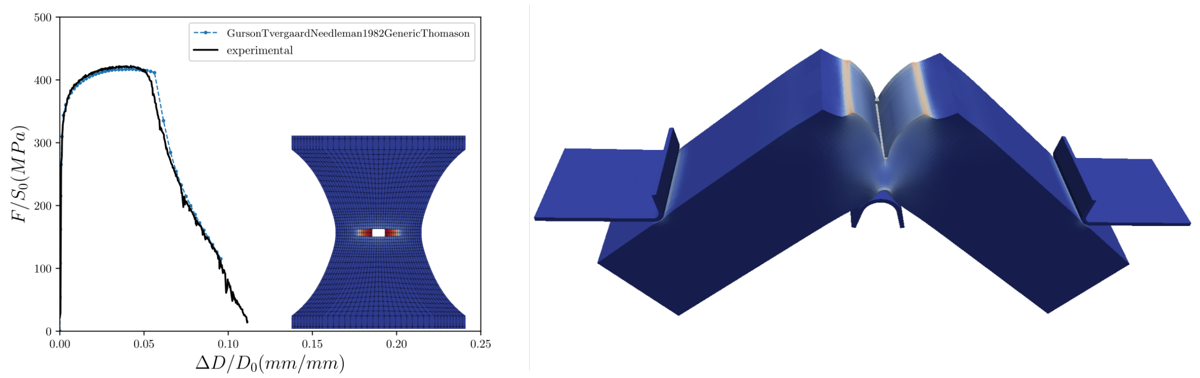

Figure 4: Extension of the `StandardElastoViscoPlasticity` brick to porous (visco-)plasticity. Application to the simulation of experimental tests with the `Cast3M` solver

- Full support for quantities, allowing mathematical objects to be associated with a unit.
- Support for tensors of arbitrary order.

2.5 Support of quantities (Version 4.0)

The term quantity denotes in `TFEL/Math` floating point numbers associated with a unit. Quantities can be used to build tensors.

Quantities allow the compiler to perform dimensional analysis at compile-time, i.e. the detection of ill-formed operations such as the addition of a strain tensor and a stress tensor.

Usage of quantities can be illustrated by the following piece of code:

```
@StateVariable strain p;
p.setGlossaryName("EquivalentPlasticStrain");

@Parameter strainrate A = 8.e-67;
@Parameter real E = 8.2;
@Parameter stress K = 1;
@Parameter stress R0 = 20e6;
@Parameter stress Rinf = 40e6;
@Parameter real bvp = 10;

@Integrator {
  ...
  // seq has the unit of a stress
  const auto seq = sigmaeq(sig);
  // iseq has the unit of the inverse of a stress
  const auto iseq = 1 / max(seps, seq);
  // the normal has no unit
  const auto n = 3 * deviator(sig) * (iseq / 2);
  // exp_bvp has no unit
  const auto exp_bvp = exp(-bvp * (p + theta * dp));
  // Rvp has the unit of a stress
  const auto Rvp = R0 + (Rinf - R0) * (1 - exp_bvp);
  if (seq > Rvp) {
    // vp has the unit of a strainrate
    const auto vp = A * pow((seq - Rvp) / K, E);
    // The residual fp has the unit a a strain (no unit)
    fp = dp - vp * dt;
    // This would not compile !
```

```
    // fp -= pow((seq - Rvp) / K, E) * dt;
    // This would not compile !
    // fp -= A * pow(seq - Rvp, E) * dt;
  }
  feel += dp * n;
}
```

2.6 Updated documentation and tutorials

The documentation of the project is constantly being improved.

The main effort has been put in developping the MFront gallery which is meant to provide detailed tutorials dedicated to the implementation of a specific behaviour. Since Version 3.2, the following tutorials have been added:

- Implementation of an isotropic linear viscous model based on Skorohold-Olevsky Viscous Sintering (SOVS) model. This page is inspired by the paper of Lester (25). For a more comprenhensive reading about the model, refer to the original paper of Olevsky (26).
- Implementation of the Fichant-La Borderie damage behaviour. See (27, 28) for a detailed description.
- Implementation of a behaviour describing the load induced thermal strain (LITS) phenomena in concrete following the work of Torelli et al. (29).
- Implementation of a behaviour coupling the Fichant-La Borderie damage behaviour and the LITS phenomena
- Implementation of the Méric-Cailletaud single crystal viscoplastic behaviour following the work of Méric et al. (30).
- Invariant-based implementation of the Mohr-Coulomb elasto-plastic model in OpenGeoSys using MFront. The algorithm used in this tutorial mostly follows the work of (31) and relies on an apex smoothing introduced by (32) and refined in (34).
- Non linear elasticity of the Ramberg-Osgood type.
- Explicit integration of the constitutive equations of a polycrystal obtained by the Berveiller-Zaoui homogeneisation scheme.
- Implementation of the `Burger_EDF_CIWAP_2021` constitutive law for concrete creep and shrinkage. The `Burger` mechanical model was originally developed for EDF R&D by (35–37). It was then modified in order to better represent long term basic creep according to (38), and implemented as a `code_aster` law and then in `MFront`. The version of the law described in this tutorial and named `Burger_EDF_CIWAP_2021` adds the desciption of the drying creep and modifies the desiccation creep law as proposed by (39).

Conclusions and future works

`MFront` is an ever improving code generation tool dedicated to material knowledge built on top of the `TFEL` project with one foot in the industrial world and one foot in the academic world.

The development of `MFront` is driven by the need of CEA and its industrial partners EDF and Framatome regarding:

- The improvement of existing legacy solvers and the development of a new generation of non linear solvers for High Performance Computing (MFEM/MGIS, MANTA, `code_aster`, etc..).
- The development of a rigourous material management process meeting the stringent requirements of the French National Nuclear Authority (ASN).

Open-source development of the project allowed numerous collaboration with industrial and academic partners illustrated by the significant number of works based on `MFront`. Significant examples are the `mgis.fenics` module and the integration of `MGIS` in `OpenGeoSys` (13). Those works significantly increase the robustness of the project and allow the exploration of state-of-the-art research which could later be integrated in industrial studies.

As a consequence, future versions will focus on the following points:

- Improvement of the documentation and addition of new tutorials.
- Various extensions of the `StandardElastoViscoplasticity` brick to simplify the implementation of the most common laws, notably in the realms of geomechanics.
- Extension of `TFEL/Math`, `MFront` and `MGIS` to support computations on GPUs.
- Coupling `MFront` with machine learning to build behaviours and models based for example on neural networks (40).
- Developments related to homogeneized behaviours.[9]
- Additional features such as support for complex initialization of the initial state of the material and built-in post-processing of the current state of the material.

Acknowledgements The authors are grateful to the many contributors to the `TFEL/MFront` project. This research was conducted in the framework of the PLEIADES project, which was supported financially by the CEA (Commissariat à l'Énergie Atomique et aux Énergies Alternatives), EDF (Électricité de France) and Framatome.

Concerning the rigorous development of a rigorous material management strategy, CEA, EDF and Framatome have formed a joined working group called `VAES` to promote best practices, gather and develop approriate tools which are largely developped around `TFEL/MFront`. This working group is a driving force for the extensions of the functionnalities of the `MFrontGallery` project.

References

1. HELFER, Thomas, FANDEUR, Olivier, GEOFFROY, Dominique, TOULEMONDE, Charles, HURE, Jérémy, DUPUY, Laurent, FORRE, Agathe, DELOISON, Dominique, PÉRALES, Frédéric, LEJEUNE, Arnaud, THIBAUD, Sébastien, RICHARD, Fabrice, GAILLARD, Yves, ALAMANSTÖTTER, Jürgen, GANGNANT, Alexandre, DRAUP, J., KECECIOGLU, Anthony, GARNIER, Christophe and ROLAND, Jérôme. New functionalities of Versions 3.1 and 3.2 of TFEL/MFront. In : *Actes du 14e Colloque National en Calcul des Structures*. Giens, France, 2019. Available from: https://hal.archives-ouvertes.fr/hal-03106120

2. MARELLE, Vincent, GOLDBRONN, Patrick, BERNAUD, Stéphane, CASTELIER, Étienne, JULIEN, Jérôme, NKONGA, Katherine, NOIROT, Laurence and RAMIÈRE, Isabelle. New developments in ALCYONE 2.0 fuel performance code. In : *Top fuel*. Boise, USA, 2016.

3. HELFER, Thomas, PROIX, Jean-Michel and FANDEUR, Olivier. Implantation de lois de comportement mécanique à l'aide de MFront : Simplicité, efficacité, robustesse et portabilité. In : *12ème colloque national en calcul des structures*. Giens, France : CSMA, June 2015.

4. HELFER, Thomas, MICHEL, Bruno, PROIX, Jean-Michel, SALVO, Maxime, SERCOMBE, Jérôme and CASELLA, Michel. Introducing the open-source mfront code generator: Application to mechanical behaviours and material knowledge management within the PLEIADES fuel element modelling platform. *Computers & Mathematics with Applications*. September 2015. Vol. 70, no. 5, p. 994–1023. DOI 10.1016/j.camwa.2015.06.027. Available from: http://www.sciencedirect.com/science/article/pii/S0898122115003132

5. STAINIER, Laurent. Introduction to MatLib: Founding concepts. 2010.

[9] *Implicit integration of the constitutive equations of a polycrystal obtained by the Berveiller-Zaoui homogeneization scheme.*. T. Helfer et al. This conference.

6. Japan Association for Nonlinear CAE. Unified material model driver for plasticity. 2018. Available from: https://www.jancae.org/annex/annexUMMDe/index.html

7. Portillo, David, Pozo, Daniel del, Rodríguez-Galán, Daniel, Segurado, Javier and Romero, Ignacio. MUESLI - a material UnivErSal LIbrary. *Advances in Engineering Software*. March 2017. Vol. 105, p. 1–8. DOI 10.1016/j.advengsoft.2017.01.007. Available from: http://www.sciencedirect.com/science/article/pii/S0965997816301430

8. Helfer, Thomas, Fandeur, Olivier, De Soza, Thomas, Deloison, Dominique and Toulemonde, Charles. Material knowledge management with the MFront code generator. Description of the ZMAT interface. Centre de l'Onéra, Chatillon, 2017. Available from: https://github.com/thelfer/tfel-doc/blob/master/Talks/ZSetUserDays2017/mfront-zset.pdf

9. Deloison, Dominique and Congourdeau, Fabrice. Testing and validation of the MFRONT interface for ABAQUS. EDF Lab Sacaly, 2016. Available from: https://github.com/thelfer/tfel-doc/blob/master/MFrontUserDays/SecondUserDay/deloison-abaqus.pdf

10. Petry, Charles and Helfer, Thomas. Advanced mechanical resolution in CYRANO3 fuel performance code using MFront generation tool. In : *LWR fuel performance meeting/TopFuel/WRFPM*. Zurich, Switzerland, July 2015.

11. Vioujard, N., Bessiron, V., Garnier, Christophe, Georget, V., Mailhé, P., Barbier, B., Deuble, D., Landskron, H., Bellanger, P. and Arimescu, V. I. GALILEO, AREVA's advanced fuel rod performance code and associated realistic methodology. In : *Proceedings of the TOPFUEL 2012 conference*. Manchester UK, 2010.

12. Helfer, Thomas, Bleyer, Jeremy, Frondelius, Tero, Yashchuk, Ivan, Nagel, Thomas and Naumov, Dmitri. The 'MFrontGenericInterfaceSupport' project. *Journal of Open Source Software*. 2020. Vol. 5, no. 48, p. 2003. DOI 10.21105/joss.02003. Available from: https://doi.org/10.21105/joss.02003

13. Bilke, Lars, Flemisch, Bernd, Kalbacher, Thomas, Kolditz, Olaf, Helmig, Rainer and Nagel, Thomas. Development of open-source porous media simulators: Principles and experiences. *Transport in Porous Media*. 1 October 2019. Vol. 130, no. 1, p. 337–361. DOI 10.1007/s11242-019-01310-1. Available from: https://doi.org/10.1007/s11242-019-01310-1

14. Perales, Frédéric, Monerie, Yann, Dubois, Frédéric, Mozul, Rémy and Babik, Fabrice. Xper : une plateforme pour la simulation numérique distribuée d'interactions multiphysiques entre corps. In : *13e colloque national en calcul des structures*. Giens, Var, France : Université Paris-Saclay, May 2017. Available from: https://hal.archives-ouvertes.fr/hal-01899269

15. Guo, N. and Yang, Z. X. NSPFEM2D: A lightweight 2D node-based smoothed particle finite element method code for modeling large deformation. *Computers and Geotechnics*. 1 December 2021. Vol. 140, p. 104484. DOI 10.1016/j.compgeo.2021.104484. Available from: https://www.sciencedirect.com/science/article/pii/S0266352X21004699

16. Dancheva, Tamara, Alonso, Unai, Barton, Michael D, Bleyer, Jeremy, Helfer, Thomas and Russo, Raffaele. Mgis.fenics part II: Cosserat media in small deformation with mgis.fenics. In : *FEniCS 2021*. Cambridge, United Kingdom, March 2021. Proceedings of FEniCS 2021. Available from: https://hal-enpc.archives-ouvertes.fr/hal-03196906

17. Mohanan, Nikhil, Bleyer, Jérémy, Helfer, Thomas and Upadhyay, Manas. Implementation of a coupled thermo-elasto-viscoplastic polycrystalline finite element model using FEniCS (mgis.fenics) and MFront. In : *7th MFront user meeting*. CEA Cadarache, France, 21 October 2021. Proceedings of the 7th MFront user meeting.

18. Camier, Jean-Sylvain. Improving performance portability and exascale software productivity with the ∇ numerical programming language. In : *Proceedings of the 3rd international conference on exascale applications and software*. Edinburgh, Scotland, UK : University of Edinburgh, 2015. p. 126–131. EASC '15. ISBN 978-0-9926615-1-9. Available from: http://dl.acm.org/citation.cfm?id=2820083.2820107

19. Bezanson, Jeff, Edelman, Alan, Karpinski, Stefan and Shah, Viral B. Julia: A fresh approach to numerical computing. *SIAM review*. 2017. Vol. 59, no. 1, p. 65–98. Available from: https://doi.org/10.1137/141000671

20. Shokeir, Mohamed, Hure, Jérémy and Helfer, Thomas. Extension of the StandardElastoViscoPlasticity brick to porous materials. 6th MFront User Meeting. 25 November 2020. Available from: https://github.com/thelfer/tfel-doc/blob/master/MFrontUserDays/SixthUserDays/Talk5-PorousPlasticity.pdf

21. Shokeir, Mohamed, Hure, Jérémy and Helfer, Thomas. Extension of the StandardElastoViscoPlasticity brick to porous materials. 2021.

22. Tvergaard, V. and Needleman, A. Analysis of the cup-cone fracture in a round tensile bar. *Acta. Metall.* 1984. Vol. 32, p. 157–169.

23. Tanguy, B. and Besson, J. An extension of the Rousselier model to viscoplastic temperature dependent materials. *Int. J. Fracture*. 2002. Vol. 116, p. 81–101.

24. Chu, C. C. and Needleman, A. Void nucleation effects in biaxially stretched sheets. *Journal of Engineering Materials and Technology*. 1980. Vol. 102, no. 3, p. 249–256.

25. Lester, Brian T. Verification of the skorohod-olevsky viscous sintering (SOVS) model. November 2017. DOI 10.2172/1411315.

26. A. Olevsky, Eugene. Sintering theory: From discrete to continuum. *Materials Science and Engineering: R: Reports*. June 1998. Vol. 23, p. 41–100. DOI 10.1016/S0927-796X(98)00009-6.

27. Fichant, Stéphanie. Endommagement et anisotropie induite du béton de structures : Modélisations approchées. thesis. Cachan, Ecole normale supérieure, 1996. Available from: http://www.theses.fr/1996DENS0015

28. Gangnant, Alexandre. Étude de la rupture quasi-fragile d'un béton à l'échelle mésoscopique : Aspects expérimentaux et modélisation. thesis. Bordeaux, 2016. Available from: http://www.theses.fr/2016BORD0401

29. Torelli, Giacomo, Mandal, Parthasarathi, Gillie, Martin and Tran, Van-Xuan. A confinement-dependent load-induced thermal strain constitutive model for concrete subjected to temperatures up to 500 °c. *International Journal of Mechanical Sciences*. 4 January 2018. DOI 10.1016/j.ijmecsci.2017.12.054. Available from: http://www.sciencedirect.com/science/article/pii/S0020740317337372

30. Méric, L. and Cailletaud, Georges. Single crystal modelling for structural calculations. *Journal of Engineering Material and Technology*. January 1991. Vol. 113, p. 171–182.

31. Nagel, Thomas, Minkley, Wolfgang, Böttcher, Norbert, Naumov, Dmitri, Görke, Uwe-Jens and Kolditz, Olaf. Implicit numerical integration and consistent linearization of inelastic constitutive models of rock salt. *Computers & Structures*. April 2017. Vol. 182, p. 87–103. DOI 10.1016/j.compstruc.2016.11.010. Available from: http://www.scopus.com/inward/record.url?eid=2-s2.0-85006482432&partnerID=MN8TOARS%20http://linkinghub.elsevier.com/retrieve/pii/S0045794916306319

32. Abbo, A. J. and Sloan, S. W. A smooth hyperbolic approximation to the Mohr-Coulomb yield criterion. *Computers & Structures*. February 1995. Vol. 54, no. 3, p. 427–441. DOI 10.1016/0045-7949(94)00339-5. Available from: http://linkinghub.elsevier.com/retrieve/pii/0045794994003395

34. Abbo, A. J., Lyamin, A. V., Sloan, S. W. and Hambleton, J. P. A C2 continuous approximation to the Mohr–Coulomb yield surface. *International Journal of Solids and Structures*. October 2011. Vol. 48, no. 21, p. 3001–3010. DOI 10.1016/j.ijsolstr.2011.06.021. Available from: http://dx.doi.org/10.1016/j.ijsolstr.2011.06.021%20https://linkinghub.elsevier.com/retrieve/pii/S0020768311002381

34. Abbo, A. J., Lyamin, A. V., Sloan, S. W. and Hambleton, J. P. A C2 continuous approximation to the Mohr–Coulomb yield surface. *International Journal of Solids and Structures*. October 2011. Vol. 48, no. 21, p. 3001–3010. DOI 10.1016/j.ijsolstr.2011.06.021. Available from: http://dx.doi.org/10.1016/j.ijsolstr.2011.06.021%20https://linkinghub.elsevier.com/retrieve/pii/S0020768311002381

35. Benboudjema, Farid. Modélisation des déformations différées du béton sous sollicitations biaxiales. Application aux enceintes de confinement de bâtiments réacteurs des centrales nucléaires. Theses. Université de Marne la Vallée, 2002. Available from: https://tel.archives-ouvertes.fr/tel-00006945

36. Benboudjema, F., Meftah, F. and Torrenti, J. M. Interaction between drying, shrinkage, creep and cracking phenomena in concrete. *Engineering Structures*. 1 January 2005. Vol. 27, no. 2, p. 239–250. DOI 10.1016/j.engstruct.2004.09.012. Available from: https://www.sciencedirect.com/science/article/pii/S0141029604003359

37. EDF. Relation de comportement BETON_BURGER_FP pour le fluage du béton. code_aster website. March 2020. Available from: https://www.code-aster.org/V2/doc/v12/fr/man_r/r7/r7.01.35.pdf

38. Sellier, Alain and Buffo-Lacarrière, Laurie. Vers une modélisation simple et unifiée du fluage propre, du retrait et du fluage en dessiccation du béton. *European Journal of Environmental and Civil Engineering*. 2009. Vol. 13, no. 10, p. 1161–1182. DOI 10.1080/19648189.2009.9693184. Available from: https://doi.org/10.1080/19648189.2009.9693184

39. Boucher, Maxime. Analyse de la corrélation spatio-temporelle des déformations entre le coeur d'un ouvrage épais et son parement : Application aux enceintes de confinement. PhD thesis. 2016. Available from: http://www.theses.fr/2016GREAI067

40. Duvillard, Marius, Giraldi, Loïc and Helfer, Thomas. Implementation of neural network based constitutive models. 7th MFront User Meeting,CEA Cadarache, 2021. Available from: https://www.researchgate.net/publication/355818919_Implementation_of_neural_network_based_constitutive_models