

## ÉLÉMENT DE PORTFOLIO 02



# Jeu de données consolidées

## 1 DÉFINITION DE CET ÉLÉMENT

**Titre de l'élément :** le Model Checking Contest

**URL de l'élément :** <https://mcc.lip6.fr>

## 2 MOTIVATIONS DU CHOIX DE CET ÉLÉMENT

Notre motivation pour faire un focus sur le Model Checking Contest (MCC) est d'illustrer un travail important de l'équipe MoVe au sein de la communauté du model checking via cet événement annuel, créé en 2011 et qui se tient tous les ans depuis (2023 verra la 13<sup>e</sup> édition se tenir à Paris en avril dans le cadre de la conférence ETAPS).

## 3 PRÉSENTATION DE CET ÉLÉMENT

Le Model Checking Contest est un concours annuel de logiciels de *model checking* dédié à l'évaluation des outils de vérification formelle pour les systèmes concurrents. Cet événement a été créé en 2011, dans la lignée d'autres concours similaires dans des disciplines voisines : SAT [1], SMT [2], CASC [8], etc.

Le MCC se focalise sur un problème particulièrement complexe : la vérification de systèmes concurrents asynchrones (au contraire de la *Hardware Model Checking competition* [4] qui concerne les systèmes synchrones). Sans être le premier du domaine, il émerge assez tôt dans le paysage des compétitions logicielles de ce type [3].

Le MCC est constitué de deux parties :

- ▶ un "appel à modèles" permettant à la communauté de soumettre des spécifications qui enrichissent chaque année un jeu de test que nous évoquerons plus loin,
- ▶ un "appel à outils" permettant aux développeurs d'outils de soumettre leur travail et de le comparer dans des conditions similaires aux autres outils compétiteurs.

### 3.1 Organisation de la compétition

Les outils sont tous soumis à 6 épreuves : calcul de l'espace d'états, calcul de propriétés structurelles, calcul de bornes sur des ensembles de places (16 formules), évaluation de 16 formules d'accessibilités, évaluation de 16 formules CTL<sup>1</sup> et évaluation de 16 formules LTL<sup>2</sup>.

Les outils peuvent bien sûr ne participer qu'à un sous-ensemble des épreuves proposées. Depuis 2014, les outils sont classés pour chacune des 6 épreuves de la compétition.

### 3.2 Un benchmark de référence consolidé

Le MCC fournit à la communauté un benchmark de référence qui est enrichi chaque année par de nouvelles contributions via l'"appel à modèles".

Ces modèles sont fournis sous la forme d'un réseau de Petri place/transition ou symétrique [5] exprimé avec le format standard PNML [6]. Ils sont vérifiés par un comité des modèles. Chacun d'eux est également fourni avec une "fiche modèle" décrivant brièvement ce qui est spécifié et expliquant, dans le cas des modèles paramétrés, le nombre d'instances du modèle et les valeurs du ou des paramètres caractérisant ces instances et leur complexité (<https://mcc.lip6.fr/models.html>).

---

1. Computational Tree Logic

2. Linear Temporal Logic

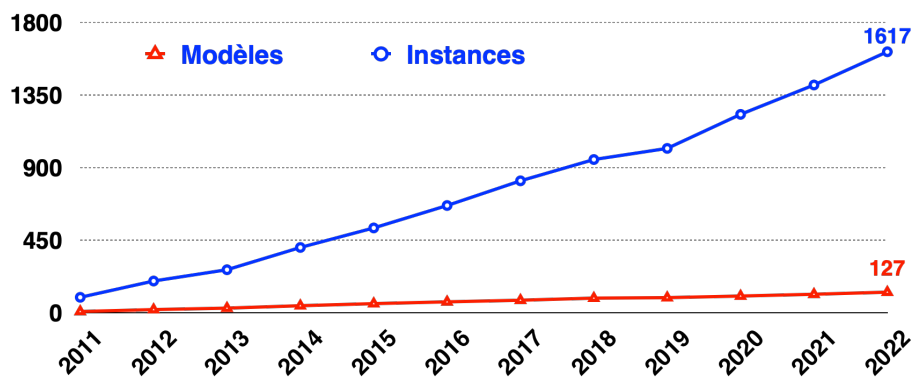


FIGURE 1 – Évolution du nombre de modèles et d'instance de modèles du MCC

La figure 1 illustre l'évolution des modèles offerts à la communauté. Leur intérêt est qu'ils proviennent de toutes origines. Ils sont issus de problèmes académiques, de cas d'études académiques réalisés dans un contexte de collaborations avec des industriels, et enfin de problèmes industriels (parfois anonymisés). Ils modélisent également des problèmes issus de domaines applicatifs très différents, de l'ingénierie des protocoles des caches de processeurs multi-cœurs, des algorithmes répartis, des systèmes biologique, etc.

De plus, pour les 4 épreuves impliquant des formules, un jeu est aléatoirement produit chaque année. Ainsi, si on considère les éditions entre 2015 (date à laquelle le format de la compétition s'est stabilisé) et 2022, ce sont 131 680 formules de chaque catégorie qui ont été produites. Pour environ la moitié d'entre elles, un résultat attendu a été calculé par une majorité qualifiée d'outils.

Les modèles et formules utilisés pour une édition du MCC sont disponibles publiquement dans une archive et les résultats sont également fournis sous la forme d'un fichier CSV consolidé. Cela a permis une analyse pluriannuelle des résultats du MCC et du comportement des outils entre 2015 et 2019 [7].


### 3.3 Bénéfices pour la communauté

Les bénéfices pour la communauté sont multiples :

- ▶ Le premier est que les outils développés par la communauté cible ont vu leur robustesse augmenter considérablement. Certains développeurs incluent des comparaisons avec les formules dont les résultats sont connus dans leur chaîne d'intégration continue.
- ▶ Le second est que certains développeurs ont pu échanger leurs techniques et partagent des modules logiciels comme, par exemple, le module transformant un réseau de Petri symétrique en un réseau place/transition équivalent.
- ▶ Le second est que le benchmark ainsi produit devient progressivement une référence. Il est utilisé (parfois de manière incomplète) et référencé dans de plus en plus d'articles produisant des résultats expérimentaux. En témoignent les 174 références au benchmark issu du Model Checking contest recensées en février 2023.

## 4 RÉFÉRENCES BIBLIOGRAPHIQUES

- [1] The international sat competition web page : <http://www.satcompetition.org>. 2022.
- [2] Clark Barrett, Leonardo de Moura, and Aaron Stump. Smt-comp : Satisfiability modulo theories competition. In *17th International Conference on Computer Aided Verification – CAV*, volume 3576 of *Lecture Notes in Computer Science*, pages 20–23. Springer, 2005.
- [3] Ezio Bartocci, Dirk Beyer, Paul E. Black, Grigory Fedyukovich, Hubert Garavel, Arnd Hartmanns, Marieke Huisman, Fabrice Kordon, Julian Nagele, Mihaela Sighireanu, Bernhard Steffen, Martin Suda, Geoff Sutcliffe, Tjark Weber, and Akihisa Yamada. Toolympics 2019 : An overview of competitions in formal methods. In Dirk Beyer, Marieke Huisman, Fabrice Kordon, and Bernhard Steffen, editors, *Tools and Algorithms for the Construction and Analysis of Systems*, pages 3–24, Cham, 2019. Springer International Publishing.
- [4] Armin Biere, Tom van Dijk, and Keijo Heljanko. Hardware model checking competition 2017. In *FMCAD*, page 9. IEEE, 2017.

- 
- [5] G. Chiola, C. Dutheillet, G. Franceschinis, and S. Haddad. A symbolic reachability graph for coloured Petri nets. *Theoretical Computer Science*, 176(1–2) :39–65, 1997.
  - [6] Lom Messan Hillah, Ekkart Kindler, Fabrice Kordon, Laure Petrucci, and Nicolas Trèves. A primer on the Petri Net Markup Language and ISO/IEC 15909-2. *Petri Net Newsletter*, 76 :9–28, October 2009.
  - [7] Fabrice Kordon, Lom Messan Hillah, Francis Hulin-Hubard, Loïg Jezequel, and Emmanuel Paviot-Adet. Study of the efficiency of model checking techniques using results of the MCC from 2015 To 2019. *International Journal on Software Tools for Technology Transfer*, June 2021.
  - [8] G. Sutcliffe. The cade-26 automated theorem proving system competition - casc-26. *AI Commun.*, 30 :419–432, 2017.