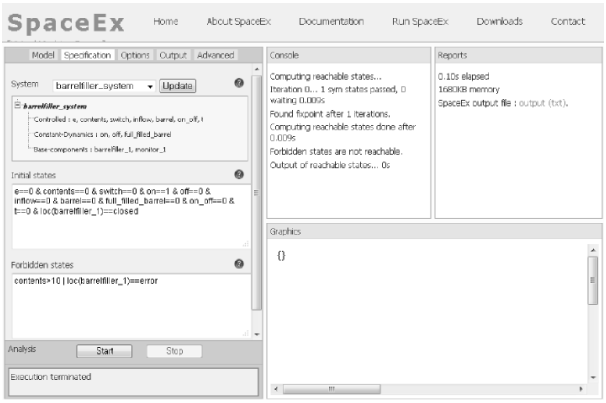

Fig. 4 Formal verification using SpaceEx

Barrel-filler system must restricts a safety property that is "*contents* must not exceeds 10 and system cannot be error state.". **Fig. 4** depicts the SpaceEx Web interfaces that shows performing safety verification of hybrid automata. We set the forbidden state as $contents > 10|loc(barrelfiller_1) = error$. *contents* is variable of barrel-filler system and must not exceeds 10. $barrelfiller_1$ is instance of linear hybrid automata for barrel-filler system. $loc(barrelfiller_1) = error$ means barrel-filler system control modes must not in error states. We set the scenario by PHAVer that verifies linear hybrid automata. The verification result is "Forbidden states are not reachable.". It means that barrel-filler system satisfies the safety requirements.

4. Conclusion

We presented a translating approach for formal verification of ECML models using hybrid automata as an example of barrel-filler system, which is input front-end of SpaceEx. ECML, a modeling language for cyber physical system, extension of DEV & DESS, models coupling between discrete and continuous elements. Hybrid automata also models combination of continuous and discrete dynamics. SpaceEx is reachability analysis tool for hybrid automata. Barrel-filler system is a suggested example for verification of ECML models. We model an ECML model for barrel-filler system and we translated from it into linear hybrid automata. We showed that ECML model can be translated into linear hybrid automata and verified using PHAVer in SpaceEx. An ECML model with non-linear dynamics has not been verified yet. We will try to verify ECML model as non-linear hybrid automata.

Acknowledgments This work was supported by the IT R&D Program of MKE/KEIT[12ND-1310, "The Development of CPS(Cyber-Physical Systems) Core Technologies for High Confidential Automatic Control Software"]. This research was supported by the MKE(The Ministry of Knowledge Economy), Korea, under the ITRC(Information Technology Research Center) support program supervised by the NIPA(National IT Industry Promotion Agency"(NIPA-2012-(H0301-12-3004))).

References

[1] Alur, R.: *Formal verification of hybrid systems*, Embedded Software (EMSOFT), 2011 Proceedings of the International Conference on, pp. 273–278 (2011).

[2] Alur, R. and Courcoubetis, C. and Halbwachs, N. and Henzinger, T.A. and Ho, P.H. and Nicollin, X. and Olivero, A. and Sifakis, J. and Yovine, S.: *The algorithmic analysis of hybrid systems* Theoretical computer science, Elsevier, Vol 138, No. 1, pp 3–34 (1995).

[3] Choi, H. and Cha, S. and Jo, J.Y. and Yoo, J. and Lee, H.Y. and Kim, W.T.: *Formal Verification of DEV&DESS Formalism Using Symbolic Model Checker HyTech*, Control and Automation, and Energy System Engineering, Springer, pp. 112–121 (2011).

[4] Frehse, G.: *PHAVer: Algorithmic verification of hybrid systems past HyTech*, Hybrid Systems: Computation and Control, Springer, pp. 258–273 (2005).

[5] Frehse, G. and Le Guernic, C. and Donzé, A. and Cotton, S. and Ray, R. and Lebeltel, O. and Ripado, R. and Girard, A. and Dang, T. and Maler, O.: *SpaceEx: Scalable Verification of Hybrid Systems*, Computer Aided Verification, Springer, pp. 379–395 (2011).

[6] Frehse, G. and Ray, R.: *Design principles for an extendable verification tool for hybrid systems*, Analysis and Design of Hybrid Systems (ADHS) (2009).

[7] Henzinger, T.A. and Ho, P.H. and Wong-Toi, H.: *HyTech: A model checker for hybrid systems*, International Journal on Software Tools for Technology Transfer (STTT), Springer, Vol. 1, No. 9, pp. 110–122 (1997).

[8] Henzinger, T.A. and Preussig, J. and Wong-Toi, H.: *Some lessons from the hytech experience*, Decision and Control, 2001. Proceedings of the 40th IEEE Conference on, IEEE, Vol. 3, pp. 2887–2892 (2001)

[9] Jeon, J. and Chun, I.G. and Kim, W.T.: *Metamodel-Based CPS Modeling Tool*, Embedded and Multimedia Computing Technology and Service, Springer, pp 285–291 (2012)

[10] Jo, J. and Yoo, J. and Choi, H. and Cha, S. and Lee, H.Y. and Kim, W.T.: *Translation from ECML to Linear Hybrid Automata*, Embedded and Multimedia Computing Technology and Service, Springer, pp. 293–300 (2012)

[11] Le Guernic, C. and Girard, A.: *Reachability analysis of hybrid systems using support functions*, Computer Aided Verification, Springer, pp. 540–554 (2009)

[12] Platzer, A. and Quesel, J.D.: *KeYmaera: A hybrid theorem prover for hybrid systems (system description)*, Automated Reasoning, Springer, pp. 171–178 (2008).

[13] Ratschan, S. and She, Z.: *HSolver: Verification of hybrid systems based on the constraint solver RSolver* (online), available from ⟨http://hsolver.sourceforge.net/⟩ (accessed 2012-07-17).

[14] Zeigler, B.P. and Praehofer, H. and Kim, T.G.: *Theory of modeling and simulation: Integrating discrete event and continuous complex dynamic systems* Academic Pr (2000).