Hierarchikus absztrakció alkalmazása állapot alapú rendszerek ellenőrzésére

Napjainkban a beágyazott rendszerek az élet minden területén egyre nagyobb teret nyernek, így helyességük ellenőrzése is egyre fontosabb, ugyanis kritikus esetben vállalatok sorsa vagy akár emberi élet is múlhat rajta. Ennek egyik fontos eszköze a formális verifikáció, melynek segítségével matematikai precizitással lehet a modellek helyességét már a tervezési fázisban vizsgálni.

A hierarchikus állapottérképek a viselkedésmodellek egyik gyakran használt eszközeként a mérnöki tervezés alapjául szolgálnak, verifikációjuk ezért kiemelt jelentőséggel bír. Gyakran azonban egy egyszerű állapottérkép ellenőrzése is nehéz feladat, ugyanis a változók számával exponenciálisan növekvő állapottér megakadályozhatja a sikeres verifikációt. Az állapottér hatékony kezelésére és bejárására az irodalomban többféle algoritmust is kidolgoztak. Ezek közül az egyik legelterjedtebb a korlátos állapotelérhetőségi analízis, amelyet leggyakrabban logikai megoldók, azaz SAT/SMT solver-ek segítségével valósítanak meg. Ehhez az állapotgépet logikai formulákkal írják le (elkódolás), majd ezeket a formulákat adják be a megoldóknak.

Gyakran azonban ezek az algoritmusok sem tudnak megbirkózni a komponensmodellekben használt változatos adattípusok és konstrukciók okozta komplex viselkedésekkel. A nagyméretű állapottér által jelentett komplexitás csökkentésére megoldást jelenthet az absztrakció alkalmazása, amely azonban elrejthet az ellenőrzés sikerességéhez elengedhetetlen részleteket. Ilyenkor finomítani kell az absztrakciót és gazdagítani a reprezentált információt. Ezen elv mentén működik az ellenpélda alapú absztrakciófinomítás (CounterExample-Guided Abstraction Refinement, CEGAR) módszere.

A gyakorlatban használt verifikációs eszközök általában nem használják ki az állapottérképekben levő hierarchiát. Dolgozatomban a célom olyan algoritmusok fejlesztése, amelyek hatékonyan tudják kezelni a hierarchikus állapottérképeket, továbbá ki tudják használni a verifikáció során a hierarchiában rejlő extra információt. Bemutatok egy általam tervezett módszert, amely lehetővé teszi komplex állapottérképek hatékony elkódolását logikai formulákká a hierarchia kihasználásával. Ezt továbbfejlesztve egy olyan absztrakciófinomításon (CEGAR) alapuló algoritmust ismertetek, amely az állapotok közötti hierarchiát felhasználja a finomítás során, és különböző logikai megoldókat felhasználva akár komplex állapottérképek ellenőrzését is lehetővé teszi. Az elkészített algoritmusok hatékonyságát egy ipari példán demonstrálom illetve hasonlítom össze.

Hierarchical Abstraction for the Verification of State-based Systems

Nowadays, as embedded systems take an increasingly important part in every aspect of our life, checking their correct behavior becomes more and more essential, especially in safety-critical cases, where a future of an enterprise or human lives rely on them. Formal verification is an important method, providing strong mathematical basis to check the correctness of the models in the design phase of the system's lifecycle.

Hierarchical statecharts, as a frequently used behavioral model, are one of the foundations of system design, so their verification has an increased relevance. However in many cases, even the verification of a simple statechart can be challenging, since the large state space can prevent the verification as it grows exponentially with the number of variables in the system. There are several algorithms in the literature to efficiently handle and explore the state space. One of the most common amongst them is the bounded state reachability analysis, which is often realized with logical solvers, such as SAT and SMT solvers. In order to perform the analysis, the transition relation of the statechart is transformed to logical formulas, and these formulas are fed to the solver.

However, even these algorithms may not handle the complex behavior caused by the various data types and constructions used in the component models. To reduce the complexity caused by the huge state space, a possible solution is to use abstraction, even though it can fade details that are inevitable for successful verification. In these cases, the abstraction needs to be refined and the represented details should be enriched. This concept is the so-called Counterexample-Guided Abstraction Refinement (CEGAR) approach.

Most of the verification techniques used in practice do not exploit the information underlying in the hierarchical structure of the statecharts. The aim of my work is to develop algorithms that can handle hierarchical statecharts efficiently, and furthermore, that can benefit from the underlying information encoded in the state hierarchy during verification. I present a newly designed approach that can be used to effectively transform complex statecharts into logical formulas, taking benefits from the hierarchy. Improving that, I introduce an algorithm based on abstraction refinement (CEGAR), that takes hierarchy information into consideration during the refinement, and makes it possible to verify complex statecharts using logical solvers. The efficiency of the previously presented algorithms is demonstrated and compared on an industrial case study.