

Chyby v překladačích programovacích jazyků mohou mít vážné následky. Kromě tradičních metod, jako je testování, je jedním ze způsobů zajištění správnosti překladačů *translation validation*, technika zajišťující sémantickou správnost optimalizací v překladačích, která se prosadila teprve v posledních pár letech. *Alive2* je open-source translation validation framework pro LLVM, který je v současnosti široce používán vývojáři LLVM. Pro zajištění použitelnosti statických analyzátorů je potřeba minimalizovat četnost falešných pozitiv. *Alive2* je navržen tak, aby neměl žádná falešná pozitiva a byl v tomto ohledu velmi úspěšný až na některé programy se smyčkami. Naším cílem v této práci je analyzovat algoritmy *Alive2* pracující se smyčkami se záměrem identifikace příčiny falešných pozitiv. Naše práce byla motivována komunikací s autory *Alive2*, kteří považovali problematiku falešných poplachů ve smyčkách jako jeden z náročnějších a závažnějších problémů v *Alive2*. Podařilo se nám přesně určit podstatu problému falešných poplachů a dokonce i opravit tento problém. Naše řešení je aktuálně součástí *Alive2*. Kromě toho jsme v *Alive2* identifikovali další potenciální problémy, o kterých v této práci také pojednáváme.