Semestrální práce z předmětu KIV/PPR

Prolomení šifry SkipJack

Zdeněk Valeš

22.11. 2018

1 Zadání

Vaším úkolem bude prolomit šifru SkipJack. Tuto šifru je výpočetně náročné prolomit hrubou silou, nicméně lze zkusit i sofistikovanější metody např. genetické a evoluční algoritmy. Abyste prolomení urychlili, lze referenční kód přepsat a vektorizovat na úrovni instrukcí, pomocí GPU, případně ho distribuovat pomocí MPI.

Samostatná práce využije alespoň dvě z celkem tří možných technologií:

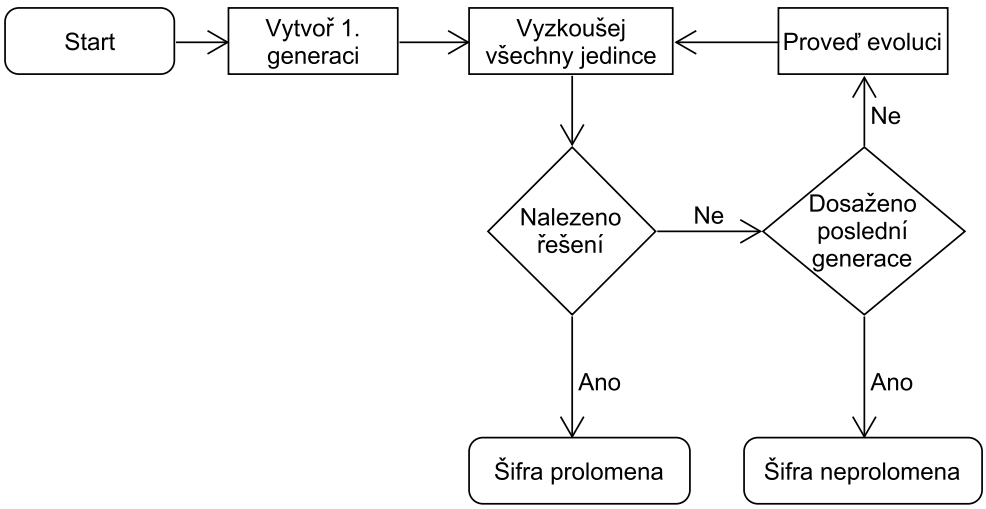
- Paralelní program pro systém se sdílenou pamětí - C++, popř. WinAPI
- Program využívající asymetrický multiprocesor - konkrétně x86 CPU a OpenCL kompatibilní GPGPU - C++ AMP
- Paralelní program pro systém s distribuovanou pamětí - C++ MPI

Práci implementujte v jazyce C++ do základu aplikace, který je k dispozici na CourseWare.

2 Popis řešení

Vstupem prolamovací funkce je zašifrovaný blok a referenční blok. Cílem algoritmu je nalezení klíče (jedince) po jehož použití v dešifrovací funkci bude vstupní blok identický s referenčním. Algoritmus je znázorněn na obrázku 1.

K prolomení šifry jsem použil diferenciální evoluci s Hammingovou vzdáleností jako cenovou funkcí. Části programu jsou paralelizované pomocí knihoven Intel TBB a OpenCL.



Obrázek 1: Algoritmus řešení

2.1 Diferenciální evoluce

Diferenciální evoluce je v principu velmi podobná klasickým genetickým algoritmům. Liší se od nich počtem rodičů (>2) potřebným k tvorbě potomka a pořadím operací mutace, křížení [1].

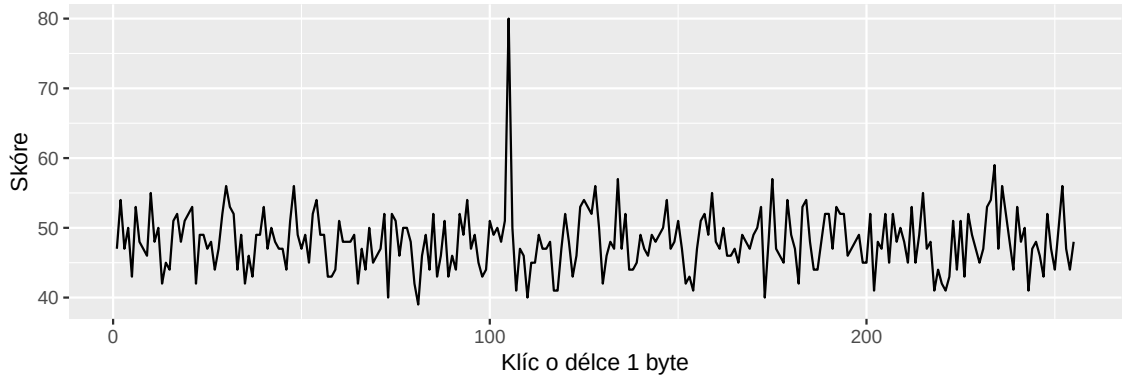
Při evoluci jedince se nejprve mutací získá šumový vektor, který se pak zkříží s aktivním jedincem. Pokud má takto vzniklý jedinec lepší skóre než aktivní, použije se do nové populace. Evoluce je řízena dvěma parametry: mutační konstantou F a prahem křížení CR. Doporučené hodnoty jsou 0,3 – 0,9 pro F a 0,8 – 0,9 pro CR [1]. V mém řešení jsem zvolil hodnoty $F = 0,3$ a $CR = 0,5$.

2.1.1 Cenová funkce

Jako cenovou funkci jsem použil variantu Hammingovo vzdálenosti dešifrovaného bloku od bloku referečního. Během implementace jsem experimentoval s více cenovými funkcemi a držel jsem se konvence větší skóre znamená lepšího jedince. U Hammingovo vzdálenosti to ale neplatí, proto jsem tuto vzdálenost odečetl od maximální možné vzdálenosti (viz rovnice 1).

$$fitness = D_{max} - D(decrypted, reference) \tag{1}$$

Vzhledem k povaze šifry SkipJack není tato cenová příliš vhodná, jak je vidět na obrázku 2. Během přednášek nám byla vyučujícím doporučena dvojrozměrná cenová funkce, tu se mi bohužel nepodařilo funkčně naimplementovat.



Obrázek 2: Hodnoty fitness funkce pro klíč délky 1

Kromě Hammingovo vzdálenosti dešifrovaného bloku od referenčního jsem experimentoval i se vzdáleností jedince od referenčního hesla. V tomto případě algoritmus konverguje dobře (100-200 generací), nicméně znalost hesla není běžný případ a proto jsem tuto cenovou funkci použil pouze k odstranění chyb evolučního algoritmu.

2.2 Paralelizace

Vektory na mutaci ne-e. Použito TBB + OpenCL, popis paralelizace evoluce (parallel_for)

3 Výsledky

4 Závěr

Reference

[1] HLAVÁČEK, Jiří. *Moderní adaptivní diferenciální evoluce* [online]. Zlín, 2015 [cit. 2018-11-22]. Dostupné z: <https://theses.cz/id/vs0ow2/>. Diplomová práce. Univerzita Tomáše Bati ve Zlíně, Fakulta aplikované informatiky. Vedoucí práce doc. Ing. Roman Šenkeřík, Ph.D