

POSSIBILITIES OF THE DISTRIBUTION OF THE GENETIC ALGORITHMS COMPUTATION

Ondřej KOTABA, Master Degree Programme (4)
Dept. of Control and Instrumentation, FEEC, BUT
E-mail: kotaba@kn.vutbr.cz

Supervised by Dr. Václav Jirsík

ABSTRACT

Describes the choices that should be made during planning phase of genetic algorithm systems design, as well as possible future aims in genetic algorithms development.

1 ÚVOD

V době, kdy se výpočetní výkon běžného domácího počítače pohybuje okolo hodnoty dvou miliard instrukcí za sekundu, vyvstává otázka, zda nepříšel čas nahradit, alespoň v některých případech, standartní postup řešení úloh - analytickými metodami - výpočtem pomocí nástroje, který dokáže sám najít alespoň takové řešení, které se ideálnímu blíží v požadovaných mezích – genetickými algoritmy.

Bohužel, pokud má existovat jakýkoliv výzkum v této oblasti, je nutné počítat dopředu se zvyšujícím se výkonem běžně dostupných počítačových systémů, a je tedy při vývoji algoritmů, které by bylo možno masově použít během několika následujících let, nutné využití aparátu, který umožní tyto provádět v čase alespoň srovnatelném s výkonem systémů, pro které je navrhován – tedy systémů mnohem rychlejších. Jedním, a v tomto případě velmi elegantním, způsobem je použít nějakou z metod distribuovaného zpracování algoritmů, nebo jejich kombinaci.

2 ROZBOR

Genetické algoritmy (GA) představují vynikající příklad jednoduše distribuovatelných algoritmů, hlavně z důvodu velmi častého opakování stejných druhů výpočtu, kdy jejich výpočetně nejnáročnější část, vyhodnocení fitness funkce, je proces, který se stejnými způsoby počítá v každé generaci, pro každého jedince a často i pro každý z prvků množiny vzorů. Platí, že v dané generaci se všechny výpočty provádí stejným způsobem přes celou množinu jedinců, v případě algoritmu s více vstupními vzory pak i přes množinu všech těchto vzorů.

3 MODEL Y DISTRIBUCE

3.1 FUNKČNÍ PARALELISMUS

Spočívá v rozdělení jednotlivých fází výpočtu GA na různé členy distribuovaného systému, tedy kód programu je rozdělen na dílčí části, nebo alespoň každý z členů používá svou příslušnou část tohoto kódu, s tím, že dílčí výsledky si navzájem předávají. Vzhledem k tomu, že GA jsou ve své podstatě rekurzivním algoritmem, je zřejmé, že pro tento model je vhodná jednosměrná kruhová struktura distribuce. Tento model nicméně není v praxi příliš využíván, u reálných GA je totiž fitness funkce mnohem náročnější na výpočet, mnohdy i o několik řádů, než operátory genetického algoritmu, není tedy v použitém kruhu rovnoměrné zatížení jednotlivých členů. Toto neplatí v případech, kdy je možné rozdělit na jednotlivé fáze výpočet samotné této funkce, neboť potom je možné rozložit zatížení sítě separací jednotlivých fází na různé členy struktury a zatížit tak síť rovnoměrně. Tato metoda bohužel vyžaduje velmi vysoký datový tok mezi jednotlivými členy, neboť jako dílčí výsledky je nutné posílat/přijímat vždy celou populaci včetně jejích fitness funkcí a to každým ze členů sítě, tolikrát za generaci, kolik je členů! Na druhou stranu může být tato varianta jednodušší na implementaci, neboť není nutný centrální bod, na které jsou v níže popsaném modelu kladeny velmi vysoké nároky.

3.2 DATOVÝ PARALELISMUS

Vzhledem k povaze GA se zde přímo nabízí možnost využití datového paralelismu, kdy kód programu, který obsahuje kromě komunikačních a jiných nezbytných rutin hlavně samotný GA, je na všechny členy distribuovaného systému přenesen celý, avšak data, na kterých tento kód provádí výpočet fitness funkce, jsou rozdělena na části a každý z členů zpracovává pouze svou část. Výhodou je zde hlavně fakt, že je v případě velkého počtu použitých vzorů či jedinců méně náročný na přenosovou kapacitu datových linek než model předchozí. Vhodná je hvězdicová struktura distribuční sítě, kdy centrální uzel buď zároveň aplikuje operátory GA, nebo toto předává jinému jejímu členu, vyhrazenému pro tuto úlohu. Nevýhodou jsou velmi vysoké nároky na komunikační schopnosti (rychlost, odezva, počet obslužitelných klientů, atd.) centrálního bodu, také musí být schopen komunikovat s mnoha členy zároveň, což přidává nutnost používat vícevláknovou strukturu centrální aplikace.

3.3 HYBRIDNÍ SYSTÉMY

Jako u mnoha jiných systémů je i zde možné použít libovolnou z kombinací uvedených distribučních modelů. Výhodnost jejich aplikace je spíše výjimkou, nicméně v některých případech, jako například třetí níže popsaný, mohou výrazně přispět k rychlosti výpočtu. Použití těchto modelů lze doporučit tam, kde je možné využít již stávající infrastrukturu počítačové sítě(sítí). Pouze jako příklad uvedu tři základní modifikace:

- Systém má standartní model kruhového funkčního paralelismu, avšak člen, který je zodpovědný za výpočet fitness funkce je sám o sobě distribuovaným systémem s libovolným z uvedených modelů.
- Systém dle modelu datového paralelismu o struktuře hvězdy, avšak každý ze členů je ve skutečnosti malý distribuovaný systém, například s kruhovou topologií – použitelné snad pouze v případě možnosti segmentace výpočtu fitness funkce.

- Překryvný model, kde jde v podstatě o kombinaci hvězdivové a kruhové topologie síťového propojení, kdy jde o kruhové zapojení s centrálním uzlem. Při tomto uspořádání lze s úspěchem rozdělit datový prostor jakýmkoliv uvedeným způsobem, dané podmnožiny při výpočtu postupně cyklují skrz jednotlivé členy na obvodu kruhu; po dokončení výpočtu s danou podmnožinou na každém okrajovém členu jsou tato data přenesena do centrálního bodu, který aplikuje operátory GA a rozešle na krajní členy příslušné podmnožiny jedinců generace nové. Bohužel, obrovskou nevýhodou této koncepce je nutnost krajních uzlů mít možnost komunikovat každý se dvěma sousedy plus s centrálním bodem a z toho vyplývající složitost synchronizace, dále potom přetrvávající nevýhoda čistě kruhového řešení – řetězový efekt, kdy výkonnost celého systému závisí na každém jedinci, jeden pomalý člen distribuční sítě zdrží všechny členy ostatní.

4 DATOVÝ PROSTOR

Soubor dat, která v danou chvíli musí být k dispozici členu distribuční sítě, sestává z těchto množin: množina jedinců v obou následných generacích a množina vzorů. Toto dává dvě hlavní možnosti rozdělení datového prostoru, v rámci jedné generace, mezi členy distribuční sítě.

- každému členu se přidělí příslušná podmnožina jedinců a celá množina vzorů. Fitness funkce, ale v tomto případě ne nutně jen ta, se potom počítá pouze pro podmnožinu jedinců a na konci výpočtu se vrátí hlavnímu uzlu, nebo jsou na ni přímo aplikovány operátory GA. Za zmínku stojí možnost nevracet výsledné jedince po každé generaci, ale místo toho použít rozdělení na, řekněme, kontinenty, kdy se daná podmnožina jedinců vyvíjí po určitý počet generací samostatně a teprve až je evoluce v rámci nich minimální, vrátit hodnoty centrálnímu uzlu a provést spojení dosažených výsledků z jednotlivých členů sítě. Otázkou zůstává, zda takový postup nebude potřebovat výrazně vyšší počet generací pro dosažení stejného stupně evoluce, než metody klasické.
- každému členu se přidělí celá množina jedinců a příslušná podmnožina vzorů. Tento potom vypočítá pouze příspěvky hodnot fitness funkce, určené z přítomné podmnožiny vzorů, což implikuje nutnost z jednotlivých příspěvků spočítat celkovou fitness funkci pro daného jedince. Tento výpočet může provádět centrální uzel, což může být náročné na jeho výpočetní výkon, nebo může být také distribuován, což je ovšem výrazně náročnější na přenosovou kapacitu komunikačních linek, neboť daná podmnožina jedinců se po dokončení výpočtu musí přenést na členu sítě, který je další v pořadí, cyklickým systémem. Tento nárůst toku dat se dá do jisté míry nahradit použitím kruhového modelu distribuční sítě.

5 ZÁVĚR

Rozhodně jsem nevyčerpal ani malé procento problémů, zajímavých řešení a nečekaných problémů, které se mohou při tvorbě, vývoji či výzkumu GA naskytnout, neboť jde o oblast, která, ač velmi zajímavá, byla v minulosti pojímána spíše za oblast čistě teoretickou a až teprve současné dění ji uvádí do praxe. Distribuci výpočtů GA považuji za, v současné době, nezbytnou. Na případnou otázku, která metoda je nejlepší, odpovím, že každá má své výhody a vždy je třeba odhadnout či vyzkoušet, která je pro daný problém nejvhodnější; neexistuje žádné pevné pravidlo.