

## ПАРАЛЕЛЬНА РЕАЛІЗАЦІЯ ГЕНЕТИЧНОГО АЛГОРИТМУ

У програмах паралельної реалізації алгоритмів розв'язок поставлених задач був обраний стиль програмування, заснований на паралелізмі задач з використанням бібліотек функцій MPI. Незважаючи на те, що такий підхід є більш трудомістким, він більш гнучкий і дозволяє різним процесам виконувати різні функції одночасно, крім того спрощується процес синхронізації процесів і управління ними.

Оскільки найбільше часу займає коригування непридатних хромосом, переклад з непрямого кодування в пряме і обчислення функції придатності, тобто знаходження оцінки кожної хромосоми, то між процесорами розподіляються саме ці операції. Реалізація цього процесу проводиться поділом популяції на рівні частини, і пересилання їх дочірнім процесам для подальшої обробки. Основним завданням дочірнього процесу є отримання масиву хромосом, його обробка, обчислення оцінки кожної хромосоми і повернення результатів батьківського процесу. В цьому випадку, якщо розмір популяції становить  $N$  хромосом і в обчисленнях беруть участь  $m$  процесорів, то на кожен з них для подальшої обробки буде передано не більше  $k = \lceil N / m \rceil + 1$  хромосома, де  $\lceil \ ]$  - ціла частина.

Кожна з реалізованих програм для паралельних обчислень має наступну ієрархічну структуру: є основний процес - "диспетчер", і кілька дочірніх, які безпосередньо займаються обчисленнями (переведенням хромосом в пряме кодування, корекцією неприпустимих хромосом і обчисленням міри придатності рішення). У свою чергу, "диспетчер" займається генерацією нових поколінь, розсилкою особин дочірнім процесам, отриманням і обробкою результатів. Хоча "диспетчер" виконує велику кількість обов'язків, все найбільш трудомісткі операції розподілені між дочірніми процесами, для досягнення найбільшої швидкодії.

Алгоритм паралельної реалізації генетичного алгоритму складається з наступної послідовності кроків:

1. Початкова ініціалізація MPI, отримання інформації про загальну кількість процесів і ID поточного процесу;
  2. Якщо поточний процес є батьківським, то зчитується інформація з вхідного файлу і розповсюджується серед дочірніх процесів;
  3. Якщо поточний процес є дочірнім, то приймається вхідна інформація;
  4. Батьківський процес генерує початкову популяцію;
  5. Батьківський процес розсилає відповідні частини популяції дочірнім процесам для обчислення оцінок хромосом;
  6. Дочірні процеси приймають частини популяції, обчислюють оцінки кожної хромосоми і повертають результати обчислень до батьківського процесу;
  7. Батьківський процес отримує результат обчислень від дочірніх процесів.
  8. Ще не оброблено достатню кількість популяцій, батьківський процес проводить турнірний відбір батьків, формує репродукційну групу і проводить операції схрещування і мутації. З усіх хромосом вибираються найкращі і саме з них формується нова популяція.
  9. Батьківський процес отримує підсумкові результати від дочірніх процесів. Завершення роботи з MPI.
- Висновок результатів.

Схема взаємодії батьківського і дочірніх процесів представлена на рис. 1.

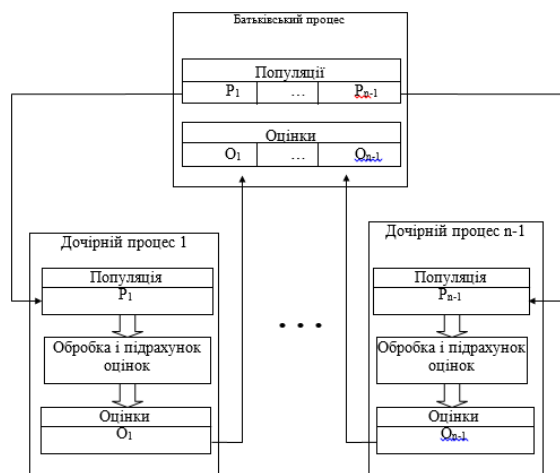


Рис. 1. Схема взаємодії дочірніх та батьківського процесів