

Plot

- Default

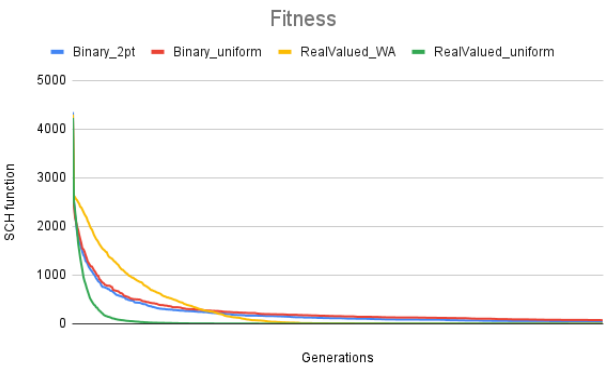


Figure 1: Fitness per generation by default setting

- Alteration to Crossover (Pc)

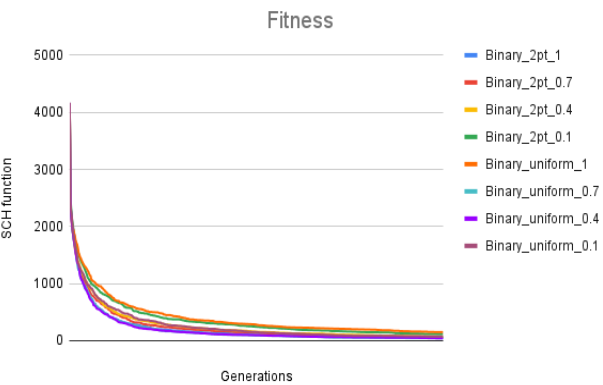


Figure 2: Fitness per generation for binary GA

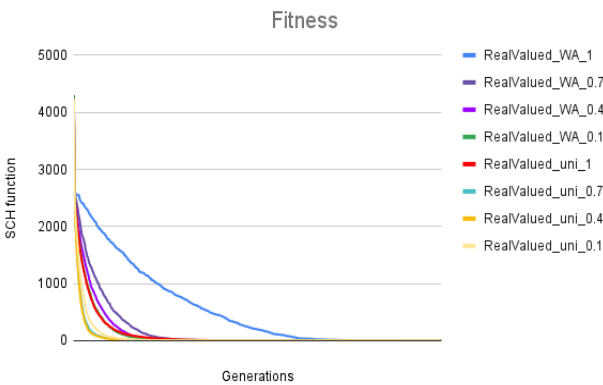


Figure 3: Fitness per generation for real-valued GA

- Alteration to Mutation (Pm)

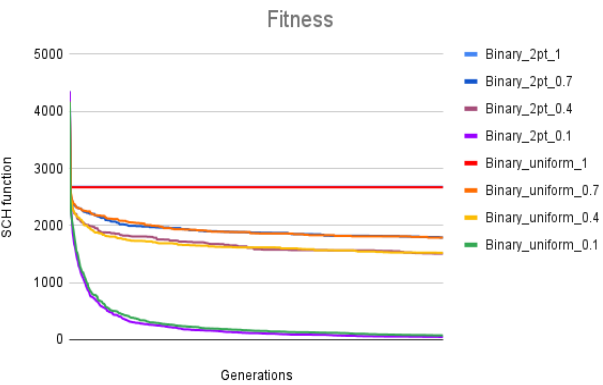


Figure 4: Fitness per generation for binary GA

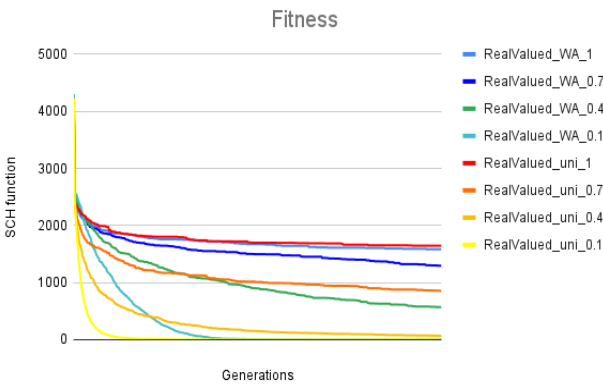


Figure 5: Fitness per generation for real-valued GA

- **Alteration to Tournament Selection (n)**

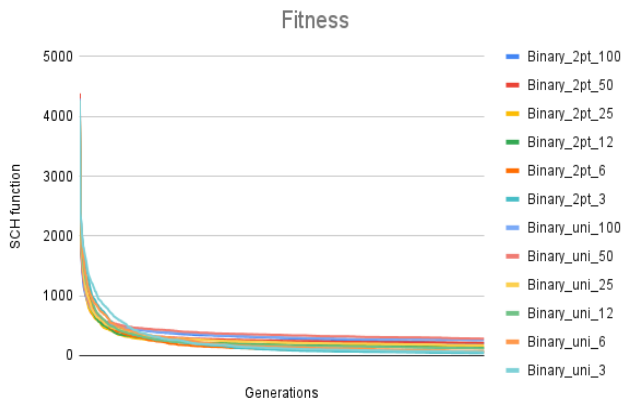


Figure 6: Fitness per generation for binary GA

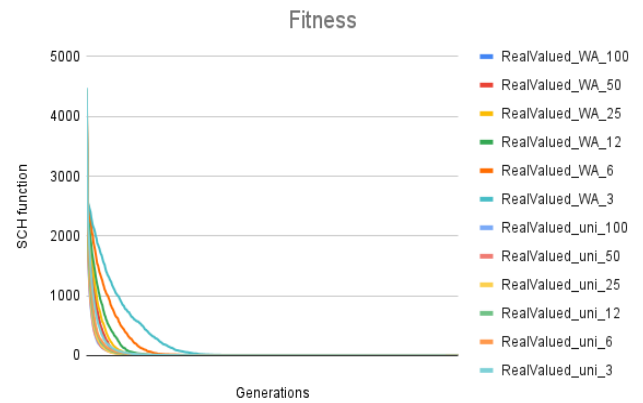


Figure 7: Fitness per generation for real-valued GA

Comparison & Reasons

- **Convergence Speed**

從Figure 1可以看出Convergence Speed為：

real-valued GA with uniform crossover > binary GA with 2-point crossover = binary GA with uniform crossover > real-valued GA with whole arithmetic crossover

在binary GA中我們可以看到使用2-point crossover或uniform crossover對於convergence speed沒什麼影響。

我想這是因為對於SCH function來說2-point crossover和uniform crossover都可以很好地保持解的多樣性；另外，SCH function在binary GA中，crossover的影響相對較小而mutation的影響相對較大。基於以上才會有第一段的結果

在real-valued GA中我們可以看到使用uniform crossover比whole arithmetic crossover收斂速度較佳。

我想是因為whole arithmetic crossover的alpha default值為0.5所以產生的子代是相同的，這樣也浪費了exploration的機會所以收斂較慢；再來是uniform crossover可以更好地維持基因的平滑性和連續性，更容易找到local optimum跟global optimum，所以收斂速度較快。

- **Solution Quality**

從Figure 1可以看出Solution Quality為：

real-valued GA with uniform crossover = real-valued GA with whole arithmetic crossover > binary GA with uniform crossover = binary GA with 2-point crossover

由於SCH function這個問題，基因的值域是實數，因此使用real_valued GA比binary GA更適合。

在real-valued GA中我們可以看到使用uniform crossover或whole arithmetic crossover最終SCH function的solution quality都很好，我想這是因為SCH function這個問題基因的值域是實數所以會有這樣的結果。

在binary GA中我們可以看到使用2-point crossover或uniform crossover對於solution quality沒什麼關係，我想是因為SCH function以及binary string的特性所以使用哪一種crossover對於最終解答都沒有太大的影響。

Try & Discussion

- **Alteration to Crossover (Pc)**

- a. binary GA with 2-point crossover**

從Figure 2可以看出binary GA with 2-point crossover中convergence speed都差不多。

我想是因為SCH function以及binary string的特性所以Pc的大小對於最終解答都沒有太大的影響。

從Figure 2可以看出binary GA with 2-point crossover中solution quality為：

$$Pc = 1 > 0.7 = 0.4 > 0.1$$

我想是因為Pc值越大可以產生新的個體就會越多，進而提升解的品質；相反地Pc值越小產生新的個體就會不夠多，進而影響解的品質。

- b. binary GA with uniform crossover**

從Figure 2可以看出binary GA with uniform crossover中convergence speed都差不多。

我想是因為SCH function以及binary string的特性所以Pc的大小對於最終解答都沒有太大的影響。

從Figure 2可以看出binary GA with uniform crossover中solution quality為：

$$Pc = 0.1 > 0.4 = 0.7 > 1$$

我想是Pc值越大在新的個體就無法保留舊的個體的特徵，可能會產生很多無用的解（過度的exploration），進而影響解的品質；相反地Pc值越小產生新的個體可以較好地保存舊個體的特徵，進而提升解的品質。

- c. real-valued GA with whole arithmetic crossover**

從Figure 3可以看出real-valued GA with whole arithmetic crossover中convergence speed為：

$$Pc = 0.1 > 0.4 = 0.7 > 1$$

我想是由於alpha值為0.5，所以若產生crossover會有兩個一模一樣的子代，這樣不利於多樣性，導致於收斂速度較慢。所以在whole arithmetic crossover且alpha=0.5時，Pc越小收斂速度越快。

從Figure 3可以看出real-valued GA with whole arithmetic crossover中solution quality都差不多。

我想這是因為SCH function的值域原本就是實數適合用real-valued GA解加上在這個問題mutation佔的角色比較重要所以Pc的大小對於最終解答都沒有太大的影響。

- d. real-valued GA with uniform crossover**

從Figure 3可以看出real-valued GA with uniform crossover中convergence speed為：

$$Pc = 0.4 = 0.7 > 0.1 > 1$$

我想這是因為Pc值過小時crossover產生的新子代可能與原先的父代過於相似（過少的exploration），導致搜索空間的探索能力不足，進而造成convergence speed下降；而Pc值過大時，新子代可能與原先的父代差距過大（過度的exploration），進而造成convergence speed下降。所以在Pc=0.4或0.7時達到了剛好的exploration，所以convergence speed最大。

從Figure 3可以看出real-valued GA with uniform crossover中solution quality都差不多。

我想這是因為SCH function的值域原本就是實數適合用real-valued GA解加上在這個問題mutation佔的角色比較重要所以Pc的大小對於最終解答都沒有太大的影響。

- **Alteration to Mutation (Pm)**

- a. binary GA**

從Figure 4可以看出在binary GA中無論使用2-point crossover或uniform crossover，得到的結果都很相似，所以合併兩種binary GA一起討論。

因為對於SCH function問題，2-point crossover和uniform crossover都可以很好地保持解的diversity，所以兩者的曲線幾乎相同。另外，由於SCH function在binary GA中，crossover的影響相對較小而mutation的影響相對較大，所以才有了第一段的結果。

從Figure 4可以看出binary GA中convergence speed為：

$$Pm = 1 > 0.7 > 0.4 > 0.1$$

我想是因為Pm越大解的多樣性越多，進而促進了解的收斂速度。

從Figure 4可以看出binary GA中solution quality為：

$$P_m = 0.1 > 0.4 > 0.7 > 1$$

我想是因為 P_m 越大解的多樣性越大，可能會引入過多的不良解，所以solution quality並不會隨著 P_m 的增加而提高，反而可能會隨著 P_m 的增加而下降。

b. real-valued GA with whole arithmetic crossover

從Figure 5可以看出real-valued GA with whole arithmetic crossover中convergence speed為：

$$P_m = 1 > 0.7 > 0.4 > 0.1$$

我想是因為 P_m 越大解的多樣性越多，進而促進了解的收斂速度。

從Figure 5可以看出real-valued GA with whole arithmetic crossover中solution quality為：

$$P_m = 0.1 > 0.4 > 0.7 > 1$$

我想是因為 P_m 越大解的多樣性越大，可能會引入過多的不良解，所以solution quality並不會隨著 P_m 的增加而提高，反而可能會隨著 P_m 的增加而下降。

c. real-valued GA with uniform crossover

從Figure 5可以看出real-valued GA with uniform crossover中convergence speed為：

$$P_m = 1 > 0.7 > 0.4 > 0.1$$

我想是因為 P_m 越大解的多樣性越多，進而促進了解的收斂速度。

從Figure 5可以看出real-valued GA with uniform crossover中solution quality為：

$$P_m = 0.1 > 0.4 > 0.7 > 1$$

我想是因為 P_m 越大解的多樣性越大，可能會引入過多的不良解，所以solution quality並不會隨著 P_m 的增加而提高，反而可能會隨著 P_m 的增加而下降。

• **Alteration to Tournament Selection (n)**

a. binary GA

從Figure 6可以看出在binary GA中無論使用2-point crossover或uniform crossover，得到的結果都很相似，所以合併兩種binary GA一起討論。

從Figure 6可以看出binary GA中convergence speed都差不多。

我想這是因為SCH function和binary GA兩者的特性。

第一個是binary GA只能表示整數，這表示個體間的差異相對於real-valued較小。第二個是SCH function並不是一個非常複雜的問題。綜合以上，當選擇壓力較小時，個體間的差異性會較大，但因為binary GA的特性，解的多樣性會增加但不會導致個體間的差異性過大，所以並不會對收斂速度產生負面的影響。當選擇壓力較大時也是同理。

從Figure 6可以看出binary GA中solution quality為：

$$n = 100 > 50 > 25 > 12 > 6 > 3$$

我想這是因為 n 值越大選擇出較好的個體的機率會越高，因此在solution quality方面可能會有所提升。

b. real-valued GA with whole arithmetic crossover

從Figure 7可以看出real-valued GA with whole arithmetic crossover中convergence speed為：

$$n = 100 > 50 > 25 > 12 > 6 > 3$$

我想這是因為 n 值越大選擇出較好的個體的機率會越高，加上whole arithmetic crossover的alpha值設為0.5時會得到兩個相同的子代，會少了一些差異性導致收斂較慢。因此若可以選擇出較好的個體，這樣得到的子代也會比較好，進而使收斂速度變快。因此 n 值越大速度會越快。

從Figure 7可以看出real-valued GA with whole arithmetic crossover中solution quality差不多。

我想這是因為在SCH function在 $N=10$ 時，在500次generation無論選擇壓力大或小都可以收斂到不錯的值。

c. real-valued GA with uniform crossover

從Figure 7可以看出real-valued GA with uniform crossover中convergence speed差不多。

我想這是因為uniform crossover可以更好地維持基因的平滑性和連續性，更容易找到local optimum跟global optimum，所以收斂速度較快，比較不會受到選擇壓力的影響。

從Figure 7可以看出real-valued GA with uniform crossover中solution quality差不多。

我想這是因為在SCH function在N=10時，在500次generation無論選擇壓力大或小都可以收斂到不錯的值。

Deal with the large-scale problem: N=100

- Default (only change N)

	binary GA with 2-point crossover	binary GA with uniform crossover	real-valued GA with whole arithmetic crossover	real-valued GA with uniform crossover
final fitness	7504.05	7458.77	15775.8	846.236

可以看見在原始設定中，兩種binary GA的表現是差不多的。

但到了real-valued GA時uniform crossover的表現比whole arithmetic crossover好，這是因為我原始的alpha設定為0.5所以會產生同樣的子代因此浪費了一次exploration的機會才會導致這個結果。

- Alteration to Pc, Pm, n, alpha, POP_SIZE

	binary GA with 2-point crossover	binary GA with uniform crossover	real-valued GA with whole arithmetic crossover	real-valued GA with uniform crossover
final fitness	6754.47	6644.27	65.0979	61.7299

binary GAs

在2-point crossover的binary GA中，從前面的觀察可以看到將Pc調大、不要改變Pm、將tournament selection n調大可以使solution quality更好。除了上述的改變，我還有適量的將POP_SIZE調大以維持更佳的diversity。

在uniform crossover的binary GA中，從前面的觀察可以看到將Pc調整在0.1、不要改變Pm、將tournament selection n調大可以使solution quality更好。除了上述的改變，我還有適量的將POP_SIZE調大以維持更佳的diversity。

real-valued GAs

在whole arithmetic crossover的real-valued GA中，我把alpha調成0.1以利產生更多樣的子代。另外，從前面的觀察可以看到不要調整Pc、將tournament selection n調大可以使solution quality更好。在Pm的部分，由於alpha已經調大，子代的多樣性已經增加，所以並不用再調整Pm。除了上述的改變，我還有適量的將POP_SIZE調大以維持更佳的diversity。

在uniform crossover的real-valued GA中，從前面的觀察可以看到將Pc調整在0.4~0.7、不要改變Pm、將tournament selection n調大可以使solution quality更好。除了上述的改變，我還有適量的將POP_SIZE調大以維持更佳的diversity。

- Alteration to #trials

	binary GA with 2-point crossover	binary GA with uniform crossover	real-valued GA with whole arithmetic crossover	real-valued GA with uniform crossover
final fitness	6420.16	6309.94	0.840433	1.19833

這是我將上面的設定多跑幾次得到的結果。

可以看到兩個binary GA都有進步但是進步幅度很小，real-valued的進步幅度就大很多。