**Updated Code to locate Single Nucleotide Polymorphisms (SNPs) and Overlapping G-Quadruplex Sequences (G4s) for Multiple Genes**

M.Shebaby

Department of Computer Science & Mathematics, School of Arts & Sciences, Lebanese American University, Byblos, email: [marc.shebaby@lau.edu](mailto:marc.shebaby@lau.edu)

**Abstract**

G-quadruplexes are non-canonical DNA secondary structures that form when guanine-rich sequences fold into a four-stranded structure stabilized by Hoogsteen base pairing. Recent studies have shown that some G-quadruplex-forming sequences can hinder DNA structure stability and can control gene regulation by impairing DNA damage repair and transcription. Single nucleotide polymorphisms (SNPs) are genetic variations that arise within DNA sequences, contributing to the genetic diversity observed between individuals. While most SNPs are non-harmful, some can be linked to various diseases. To investigate the potential role of SNPs in causing G-quadruplexes, we developed a new Python code that automatically computes the distance between SNPs and the closest G-quadruplex complexes for multiple genes. Additionally, the code integrates the G4 Hunter tool to detect G-quadruplex sequences in genes with high scores. Overall, the updated code provides a more comprehensive approach to identifying the overlaps between SNPs and G-quadruplex sequences which could lead to a better understanding of the possible significant role of SNP in the formation/maintenance of G-quadruplex structures in disease development.

**Keywords**: G4-quadruplex, G4 sequence, SNP (single nucleotide polymorphism), G4Hunter tool.

1. **Introduction**

Small genetic variations in the form of single nucleotide polymorphisms (SNPs) are highly prone to occur within nucleic acids of the double helix DNA, which can significantly impact gene regulation and protein function. While SNPs are common and crucial for genetic diversity, some can lead to clinical manifestations [7, 11]. In addition, the planar arrangement of four guanine bases linked by non-Watson-Crick hydrogen bonds, known as G-quadruplexes, can stabilize the secondary structure of DNA and also contribute to clinical manifestations. The discovery of the guanine tetrad-forming sequence motif was a result of early studies on the self-assembly of guanylic acid, where double helix B-DNA form can fold into G-quadruplexes [5,10]. There is a noteworthy correlation between SNPs found in G-quadruplex sequences and the expression levels of the corresponding gene [1]. Furthermore, it was discovered that somatic mutations situated in the 5’ untranslated regions (UTR) modify the stability of RNA G4 and consequently have an impact on gene expression in cancer patients [12]. As the topic of G4-quadruplexes gains significant attention, numerous methods have been developed to computationally predict the locations of G4 quadruplexes, with G4Hunter being one of the most recent ones. This algorithm uses a sliding window approach and takes into account the G-richness (the fraction of Gs in a sequence) and G-skewness (the G/C asymmetry between the complementary strands) to identify regions with a mean score above a certain threshold. Empirical evidence demonstrated the high accuracy and sensitivity of this approach compared to other methods that deal with only G-richness and a consensus sequence for G4-quadruplex [2]. Recent literature began to rely extensively on the G4Hunter tool in an attempt to explore the functionality of the G4-quadruplexes [3,4]. Nevertheless, there has not been any published work that integrates this tool to identify overlaps of SNPs and G4-quadruplexes and contribute to the topic of the association between them. Thus, this work provides a Python script that automatically maps the distances of SNPs of various genes to their closest G4 sequences predicted by the G4Hunter tool.

1. F**unctional Implications of G4-Quadruplex on DNA**

Experimental studies have shown that G4-quadruplex structures can obstruct the progress of the DNA replication or transcription machinery, resulting in a stalled fork [6]. This, in turn, facilitates the creation of one-ended breaks in the chromosome, increasing its vulnerability to rearrangements and instability [8]. G4-quadruplex structures occurring at noncoding regions, such as the 5' UTR of RNA, have been demonstrated to affect regulatory gene expression as translational repressors in certain proto-oncogenes of cancer patients. Moreover, the discovery of specific SNPs near those G4 sequences can have a significant impact on the stability of their structures, thereby influencing their regulatory function. In a patient with melanoma, SNPs overlapping a predicted RNA G4 lead to its destabilization and to an augmented translation efficiency in the CXCL14 gene (tumor suppressor gene). On the other hand, the TAOK21 gene a tumor suppressor gene, responsible for inducing apoptosis, becomes downregulated due to the stabilization of the G4 structures by overlapping SNPs [12]. Furthermore, emerging evidence suggests a potential link between G4-quadruplex structures and neurological disorders. For example, in the case of Fragile X syndrome, the expansion of a CGG repeat (G4 motif) in the FMR1 gene is associated with a genetic disorder characterized by intellectual disability and various behavioral and physical features [8].

1. **G4Hunter**

G4Hunter is one of the most prominent predictive tools for G4-quadruplexes, where it accounts for G richness and G skewness of a DNA sequence by a sliding window-based scoring algorithm. Each position in the sequence is given a score between -4 and 4, with A and T receiving a score of 0 (neutral), and G receiving a plus score. G-rich regions are given higher scores according to the scoring system, which gives single Gs a value of 1, GG sequences a score of 2, GGG sequences a score of 3 and sequences with four or more Gs a score of 4. G4Huter is able to simultaneously score the complementary strand as well by providing Cs with negative and similar values as Gs. G4Hunter score is computed as the arithmetic mean of this ‘sequence’ of integers inside the range of the window.

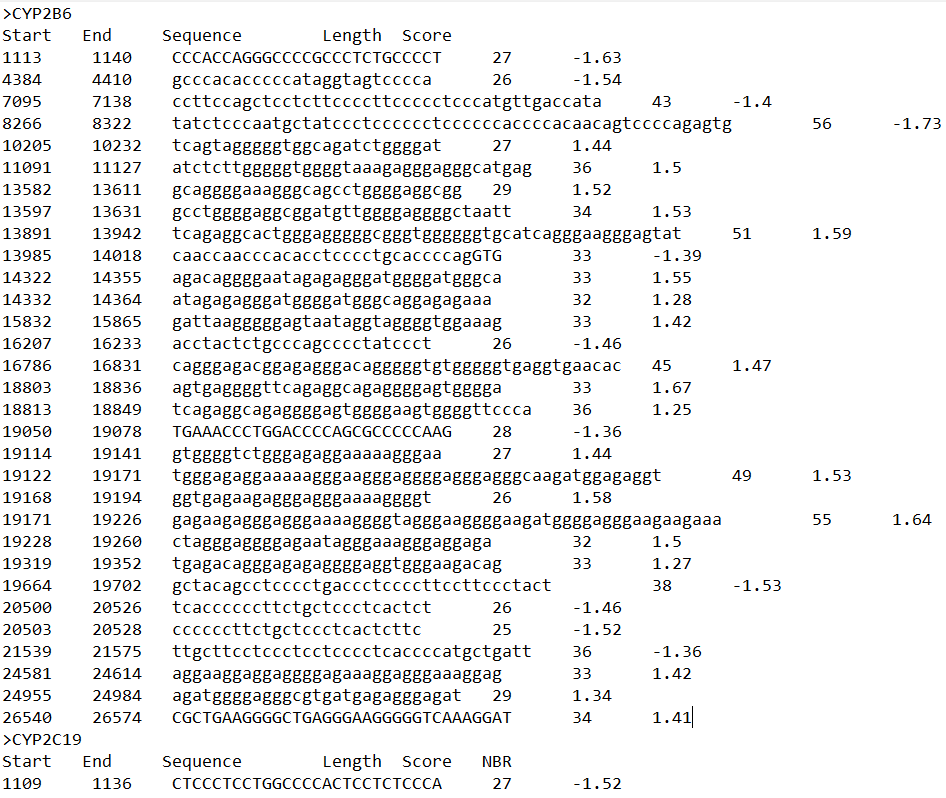
1. **Procedure**

Python program version 3.9.7 or later is preferred to be installed along with the libraries [to run the code. This work expands the functionality of our previous two codes “SNP-locator” and “G4-overlap” [9] and eliminates the need for a user manual.

* 1. **Workflow**

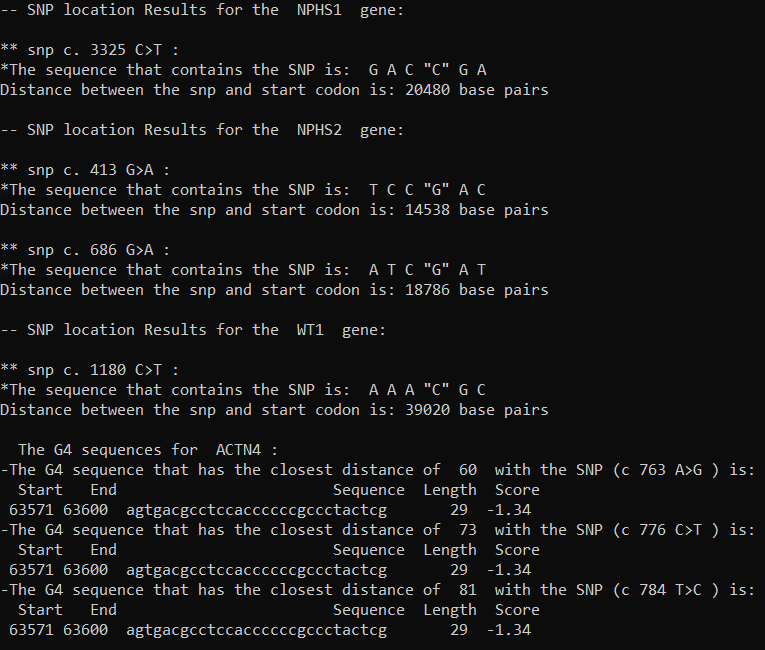
The developed code first locates the distances of SNPs relative to the start codon and their respective positions (flanking region) in the genomic file. This is achieved by calling a function that opens the input directory provided by the user to read each Fasta file individually. Moreover, as the files are being read, the start codon is detected, and the number of base pairs in the exon region is tracked. Once this number reaches the SNP position in the coding DNA (c.DNA), the function checks if there is a particular distance upstream or downstream of the located position that needs to be computed and adjusts the total distance of the SNP accordingly. The function repeats this step for all the SNPs of the genes available and stores the results in a dictionary.

As the code incorporates the G4Hunter tool with some modifications to allow it to run on multiple Fasta files, a new text file called “-G4\_Merged.txt” will be automatically created in the directory provided by the user which contains the predicted G4 sequences of the genes with their start position, end position, length, and scores (Figure 1).



**Figure 1.** The G4 sequences predicted by the G4Hunter tool with their positions (start and end), length, and scores

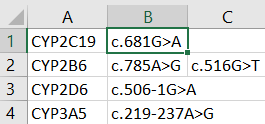
Another function in the code stores the locations of G4 sequences in a dictionary while reading the “-G4\_Merged.txt” file and performs two consecutive computations to map the sequences with the SNPs. First, it calculates the distance between G4 sequences and the start codon. Afterward, it measures the distance between each G4 sequence and the SNPs. This is achieved by either subtracting the end position of the G4 sequence from the SNP position if the SNP is situated downstream of the G4 sequence or by subtracting the SNP position from the start position of the G4 sequence if the SNP is located upstream of the G4 sequence. The same process is repeated for every provided gene and the results are stored in a list. Finally, the SNPs’ positions in the genomic file and their closest G4 sequences, along with their characteristics, are returned in the output as shown in Figure 2.



**Figure 2.** Representation of the output of the code; SNP location of a gene (NPHS1and WT1 at the top) and G4 sequences results for a gene (ACTN4 at the bottom)

* 1. **Data Entry**

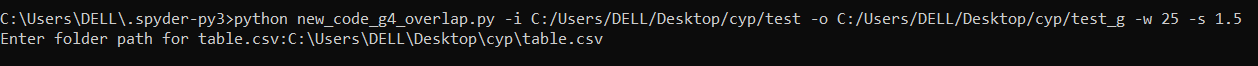
In order for the code to execute properly there exist some guidelines on the input data. Fasta files with 1000 nucleobase pairs before the promoter are accepted, however, they have to include capital letters for the bases in the exon regions and small letters for the bases in the intron regions. Such files can be obtained from the UCSC Genome Browser, where each file represents one whole gene. Furthermore, the code takes the path of the CSV file that contains the SNPs of the genes in the format shown in Figure 3. It is important that the locations of the selected SNPs are in the cDNA and that all of the Fasta files are in the same directory.



**Figure 3.** Format of the CSV file with the SNPs of the genes.

* 1. **Instructions to run the code**

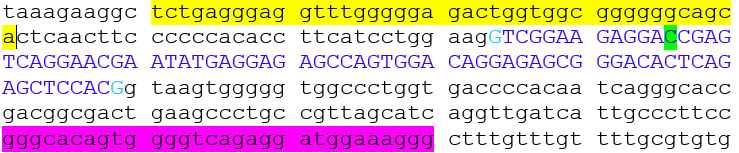
To run the code, the terminal has to be opened in the same directory as that of the code. Then, the following command can be passed: ("python <code\_name.py> -i <inputrepository> -o <outputrepository> -w <window size> -s <score threshold>") and the user will be prompted to enter the direct path to the CSV file (Figure 4). After entering the CSV file path, the code will be executed, generating a file that contains the predicted G4 sequences in the output repository, and displaying the results in the terminal.



**Figure 4.** The input of the code; the path of the fasta files as input repository, the path of the folder for output repository, window size, threshold value, and the path of the CSV file that contain the SNPs.

1. **Summary and Validation**

In summary, this code was developed to contribute to studies focused on discovering the functionality of G4 quadruplexes, specifically by detecting G4 sequences that overlap or are in close proximity to single nucleotide polymorphisms (SNPs); these SNPs may potentially serve as factors for either stabilizing and forming G4 structures or destabilizing them. Moreover, it allows future studies to analyze specific G4 sequences for various gene expressions and their corresponding diseases, eliminating the need to examine a broad range of G4 sequences, which can be time-consuming. Furthermore, when tested on the same set of genes, the new code exhibited similar results for SNP locations and enhanced performance in detecting G4 sequences in proximity to SNPs, as compared to our previous study [9]. Notably, Figure 5 and Table 1 showcase the new code's capability to accurately identify a G4 sequence directly adjacent to the SNP within the genes. Table 1, displays 10 genes out of 15 genes that possess possible G4 sequences with a distance of less than 500 base pairs from the SNPs. Nevertheless, in this work, the code was executed only for a small number of genes and did not include any new sightings. Therefore, to thoroughly investigate the frequency of the closeness or overlaps of G4 sequences with SNPs, future studies can implement this code with a larger set of genes from various biological databases.



**Figure 5.** Genomic sequence of the NPHS1 gene. The G4 sequence highlighted in yellow was detected by the new code to have the closest distance with the SNP (highlighted in green; c.3325C>T). Whereas, the old code detects the G4 sequence highlighted in purple as the closest to the SNP.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Gene | Start | End | Sequence | Length | Score | Distance from the SNP |
| ACTN4 | 63571 | 63600 | agtgacgcctccaccccccgccctactcg | 29 | -1.34 | 60 |
| CYP2A6 | 2636 | 2662 | cccctgccgccccctggcctgtctcc | 26 | -1.62 | 62 |
| CYP2B6 | 19050 | 19078 | TGAAACCCTGGACCCCAGCGCCCCCAAG | 28 | -1.36 | 0 |
| CYP2C8 | 22992 | 3030 | agctttggtaactggggtgagggggatggaaaacagag | 38 | 1.18 | 195 |
| CYP2C9 | 4390 | 4423 | tctgtcttggggatggggaggatggaaaacaga | 33 | 1.18 | 210 |
| CYP2D6 | 2846 | 2886 | cccgcatctcccacccccagGACGCCCCTTTCGCCCCAAC | 40 | -1.7 | 0 |
| ITGA3 | 21582 | 21607 | ccttcacacctccggccacccccca | 25 | -1.52 | 60 |
| ITGB4 | 31779 | 31804 | GGGGCCGGCTGGGGGCCTGAGCGGG | 25 | 1.64 | 65 |
| NPHS1 | 21551 | 21590 | tgagggaggtttgggggagactggtggcggggggcagca | 39 | 1.64 | 46 |
| NPHS2 | 20210 | 20258 | tggcaggaacggtggggttggtggggatggacaggaggggttggtaca | 48 | 1.5 | 339 |

**Table 1.** 10 out of 15 genes had SNPs with a distance of less than 500 base pairs from the predicted G4 sequence; distance=0 indicates overlaps. G4 sequences extracted by the code are either found in the same strand of the provided genomic sequence if the score >0 or in the opposite strand if the score <0.

**Availability and implementation**

The code can be found at https://github.com/Marc-shebaby/Capstone-Project.git.

1. **References**

1. Baral, A., Kumar, P., Halder, R., Mani, P., Yadav, V. K., Singh, A., Das, S. K., & Chowdhury, S. (2012). Quadruplex-single nucleotide polymorphisms (Quad-SNP) influence gene expression difference among individuals. Nucleic acids research, 40(9), 3800–3811.

2. Bedrat, A., Lacroix, L., & Mergny, J. L. (2016). Re-evaluation of G-quadruplex propensity with G4Hunter. Nucleic acids research, 44(4), 1746–1759.

3. Brázda, V., Luo, Y., Bartas, M., Kaura, P., Porubiaková, O., Šťastný, J., Pečinka, P., Verga, D., Da Cunha, V., Takahashi, T. S., Forterre, P., Myllykallio, H., Fojta, M., & Mergny, J. L. (2020). G-Quadruplexes in the Archaea Domain. Biomolecules, 10(9), 1349.

4. Cantara, A., Luo, Y., Dobrovolná, M., Bohalova, N., Fojta, M., Verga, D., Guittat, L., Cucchiarini, A., Savrimoutou, S., Häberli, C., Guillon, J., Keiser, J., Brázda, V., & Mergny, J. L. (2022). G-quadruplexes in helminth parasites. Nucleic acids research, 50(5), 2719–2735.

5. Davis J. T. (2004). G-quartets 40 years later: from 5'-GMP to molecular biology and supramolecular chemistry. Angewandte Chemie (International ed. in English), 43(6), 668–698.

6. Kruisselbrink, E., Guryev, V., Brouwer, K., Pontier, D. B., Cuppen, E., & Tijsterman, M. (2008). Mutagenic capacity of endogenous G4 DNA underlies genome instability in FANCJ-defective C. elegans. Current biology : CB, 18(12), 900–905.

7. Löwik, M. M., Groenen, P. J., Levtchenko, E. N., Monnens, L. A., & van den Heuvel, L. P. (2009). Molecular genetic analysis of podocyte genes in focal segmental glomerulosclerosis--a review. *European journal of pediatrics*, *168*(11), 1291–1304.

8. Maizels N. G4-associated human diseases. EMBO Rep. 2015 Aug;16(8):910-22. doi: 10.15252/embr.201540607. Epub 2015 Jul 6. PMID: 26150098; PMCID: PMC4552485.

9. Saad, M., Shebaby, M., Mehawej, C., & Faour, W. (2022). New complementary python codes to locate Single Nucleotide Polymorphisms (SNPs) and Overlapping G-Quadruplex Sequences (G4s). MethodsX, 9, 101875.

10. Sen, D., & Gilbert, W. (1988). Formation of parallel four-stranded complexes by guanine-rich motifs in DNA and its implications for meiosis. *Nature*, *334*(6180), 364–366.

11. Sim, S.C., Ingelman-Sundberg, M. The Human Cytochrome P450 (CYP) Allele Nomenclature website: a peer-reviewed database of CYP variants and their associated effects. Hum Genomics 4, 278 (2010).

12. Zeraati, M., Moye, A. L., Wong, J. W. H., Perera, D., Cowley, M. J., Christ, D. U., Bryan, T. M., & Dinger, M. E. (2017). Cancer-associated noncoding mutations affect RNA G-quadruplex-mediated regulation of gene expression. Scientific reports, 7(1), 708. (10)