ID 53

Computational investigations in eukaryotes genome de novo assembly using short reads

Leandro Cintra, CNPTIA - Embrapa Informática Agropecuária

Recently news technologies in molecular biology enormously improved the sequencing data production, making it possible to generate billions of short reads totalizing gibabases of data per experiment. Prices for sequencing are decreasing rapidly and experiments that were impossible in the past because of costs are now being executed. Computational methodologies that were successfully used to solve the genome assembler problem with data obtained by the shotgun strategy, are now inefficient. Efforts are under way to develop new programs. At this moment, a stabilized condition for producing quality assembles is to use paired-end reads to virtually increase the length of reads, but there is a lot of controversy in other points. The works described in literature basically use two strategies: one is based in a high coverage[1] and the other is based in an incremental assembly, using the made pairs with shorter inserts first[2]. Independently of the strategy used the computational resources demanded are actually very high. Basically the present computational solution for the de novo genome assembly involves the generation of a graph of some kind [3], and one because those graphs use as node whole reads or k-mers, and considering that the amount of reads is very expressive; it is possible to infer that the memory resource of the computational system will be very important. Works in literature corroborate this idea showing that multiprocessors computational systems with at least 512 Gb of principal memory were used in de novo projects of eukaryotes [1,2,3]. As an example and benchmark source it is possible use the Panda project, which was executed by a research group consortium at China and generated de novo genome of the giant Panda (Ailuropoda melanoleura) . The project initially produced 231 Gb of raw data, which was reduced to 176 Gb after removing low-quality and duplicated reads. In the de novo assembly process just 134 Gb were used. Those bases were distributed in approximately 3 billions short reads. After the assembly, 200604 contigs were generated and 5701 multicontig scaffolds were obtained using 124336 contigs. The N50 was respectively . 36728 bp and 1.22 Mb for contigs and scaffolds. The present work investigated the computational demands of de novo assembly of eukaryotes genomes, reproducing the results of the Panda project. The strategy used was incremental as implemented in the SOAPdenovo software, which basically divides the assembly process in four steps: pre-graph to construction of kmer-graph; contig to eliminate errors and output contigs, map to map reads in the contigs and scaff to scaffold contigs. It used a NUMA (non-uniform memory access) computational system with 8 six-core processors with hyperthread tecnology and 512 Gb of RAM (random access memory), and the consumption of resources as memory and processor time were pointed for every steps in the process. The incremental strategy to solve the problem seems practical and can produce effective results. At this moment a work is in progress which is investigating a new methodology to group the short reads together using the entropy concept. It is possible that assemblies with better quality will be generated, because this methodology initially uses more informative reads. References [1] Gnerre et. al.; High-quality draft assemblies of mammalian genomes from massively parallel sequence data, Proceedings of the National Academy of Sciences USA, v. 108, n. 4, p. 1513-1518, 2010 [2] Li et. al.; The sequence and de novo assembly of the giant panda genome, Nature, v. 463, p. 311-317, 2010 [3] Schatz et. al.; Assembly of large genomes using second-generation sequencing, Genome Research, v. 20, p. 1165-1173, 2010