CLAIMS

Claim 1. A method for hardware-accelerated generation of a K-mer graph using a programmable logic device, the method comprising:

obtaining a first set of nucleic acid sequences, wherein the first set of nucleic acid sequences includes (i) a plurality of reads corresponding to a region of a reference sequence and (ii) a portion of the reference sequence;

generating, using a plurality of non-pipelined hardware logic units of a programmable logic device a K-mer graph using the obtained first set of nucleic acid sequences, wherein each hardware logic unit comprises a different hardware logic circuit configured to perform one or more operations, wherein each node of the K-mer graph represents a K-mer, each edge of the K-mer graph represents a link between a pair of K-mers, and each weight of each edge of the K-mer graph represents a number of occurrences of a K-mer sequence represented by a pair of K-mers; and

during generation of the K-mer graph:

periodically updating, with a control machine, graph description data for the K-mer graph after performance of the one or more operations by each hardware logic unit that is used to generate at least a portion of the K-mer graph, wherein the graph description data represents (i) a K-mer graph identifier and (ii) K-mer graph state information, wherein the control machine creates a workflow of operations using the non-pipelined hardware logic units by triggering performance of the one or more operations of each respective hardware logic unit during generation of the K-mer graph.

Claim 2. The method of claim 1, wherein the output of each hardware logic unit of the plurality of hardware logic units is stored via a hash table cache.

Claim 3. The method of claim 1, wherein the control machine is implemented using a hardware logic unit of the programmable logic device.

Claim 4. The method of claim 1, wherein the control machine is implemented by using one or more CPUs or GPUs to execute software instructions to realize functionality of the control machine.

Claim 5. The method of claim 1, the operations further comprising:

providing the generated K-mer graph to a variant calling unit, wherein the variant calling unit processes the K-mer graph to determine candidate variants between one or more of the plurality of reads and the reference sequence.

Claim 6. The method of claim 5, wherein software instructions are executed by one or more CPUs or GPUs to realize one or more functions of the variant calling unit.

Claim 7. The method of claim 5, wherein the programmable logic device is used to accelerate one or more functions of the variant calling unit.

Claim 8. The method of claim 1, wherein the graph description data further includes (iii) data representing a last hardware logic unit of the plurality of hardware logic units that executed hardware logic on the K-mer graph or nucleic acid sequences of a pileup associated with the K-mer graph identifier.

Claim 9. A system for hardware-accelerated generation of a K-mer graph using a programmable logic device, the system comprising:

a hardware-accelerated graph generation unit that includes hardware digital logic circuits that have been arranged to perform operations, the operations comprising:

obtaining a first set of nucleic acid sequences, wherein the first set of nucleic acid sequences includes (i) a plurality of reads corresponding to a region of a reference sequence and (ii) a portion of the reference sequence;

generating, using a plurality of non-pipelined hardware logic units of a programmable logic device a K-mer graph using the obtained first set of nucleic acid sequences, wherein each hardware logic unit comprises a different hardware logic circuit configured to perform one or more operations, wherein each node of the K-mer graph represents a K-mer, each edge of the K-mer graph represents a link between a pair of K-mers, and each weight of each edge of the K-mer graph represents a number of occurrences of a K-mer sequence represented by a pair of K-mers; and

during generation of the K-mer graph:

periodically updating, with a control machine, graph description data for the K-mer graph after performance of the one or more operations by each hardware logic unit that is used to generate at least a portion of the K-mer graph, wherein the graph description data represents (i) a K-mer graph identifier and (ii) K-mer graph state information, wherein the control machine creates a workflow of operations using the non-pipelined hardware logic units by triggering performance of the one or more operations of each respective hardware logic unit during generation of the K-mer graph.

Claim 10. The system of claim 9, wherein the output of each hardware logic unit of the plurality of hardware logic units is stored via a hash table cache.

Claim 11. The system of claim 9, wherein the operations further include:

providing the generated K-mer graph to a variant calling unit, wherein the variant calling unit is configured to process the K-mer graph to determine candidate variants between one or more of the plurality of reads and the reference sequence.

Claim 12. The system of claim 9, the system further comprising:

one or more computers; and

one or more memory devices storing instructions that, when executed by the one or more computers, cause the one or more computers to perform second operations of a variant calling unit, the second operations comprising

obtaining, by the variant calling unit, the generated K-mer graph; and

identifying, based on the variant calling unit processing the generated K-mer graph, one or more candidate variants, wherein a candidate variant is a difference between a base call of one or more reads in a pileup of reads and a nucleotide of a reference genome at a particular location of the reference genome.

Claim 13. The system of claim 9, wherein the operations further include:

obtaining, by a variant calling unit, the generated K-mer graph; and

identifying, based on the variant calling unit processing the generated K-mer graph, one or more candidate variants, wherein a candidate variant is a difference between a base call of one or more reads in a pileup of reads and a nucleotide of a reference genome at a particular location of the reference genome.

Claim 14. The system of claim 9, wherein the graph description data further includes (iii) data representing a last hardware logic unit of the plurality of hardware logic units that executed hardware logic on the K-mer graph or nucleic acid sequences of a pileup associated with the K-mer graph identifier.

Claim 15. A method for hardware-accelerated generation of a K-mer graph in a programmable logic device, the method comprising:

obtaining a first set of nucleic acid sequences, wherein the first set of nucleic acid sequences include (i) a plurality of reads corresponding to a region of a reference sequence and (ii) a portion of the reference sequence;

for each particular nucleic acid sequence of the first set of nucleic acid sequences:

generating, for storage in a hash table cache and by a first hardware logic unit, data representing a graph node for each K-mer of the particular nucleic acid sequence;

detecting, by a control machine, that the first hardware logic unit has completed generation of a graph node for each K-mer of the particular nucleic acid sequence;

configuring, by the control machine, a second hardware logic unit to perform graph edge generation for the generated graph nodes; and

for one or more pairs of the generated graph nodes:

generating, by the second hardware logic unit and for storage in the graph hash table, data representing graph edges between one or more pairs of the generated graph nodes generated by the first hardware logic unit, wherein the data representing the graph node for each K-mer stored in the hash table cache and the data representing graph edges stored in the hash table cache represents a K-mer graph of the first set of nucleic acid sequences.

Claim 16. The method of claim 15, the method further comprising:

periodically storing, by the control machine and in a memory unit that is accessible by the control machine, graph description data for an instance of the K-mer graph, wherein the graph description data represents (i) a K-mer graph identifier and (ii) K-mer graph state information.

Claim 17. The method of claim 15,

wherein the first hardware logic unit is further configured to:

determine whether one or more of the particular K-mers of the particular nucleic acid sequence matches another K-mer of the particular nucleic acid sequence; and

based on a determination that the one or more particular K-mers of the particular nucleic acid sequence match another K-mer of the particular nucleic acid sequence, store data that marks the one or more particular K-mers as non-unique K-mers.

Claim 18. The method of claim 15, wherein the second hardware logic is further configured to:

assign an edge weight to each edge of the K-mer graph.

Claim 19. The method of claim 15, the method further comprising:

instructing a third hardware logic unit of the programmable logic device to execute hardware logic configured to:

obtain data representing the K-mer graph from the hash table cache; and

provide the obtained data representing the K-mer graph to a variant calling unit.

Claim 20. The method of claim 15, the method further comprising:

instructing a third hardware logic unit of the programmable logic device to execute hardware logic configured to:

selectively delete data representing graph nodes and data representing graph edges of the K-mer graph from the hash table cache.

Claim 21. The method of claim 15, wherein the control machine is implemented using a third hardware logic unit of the programmable logic device.

Claim 22. The method of claim 15, wherein the hash table cache is implemented using a third hardware logic unit of the programmable logic device.

Claim 23. The method of claim 15, wherein the control machine is implemented using one or more CPUs or GPUs that execute software instructions to realize the functionality of the control machine.

Claim 24. The method of claim 15, wherein the graph description data further include (iii) data representing a last hardware logic unit of the plurality of hardware logic units that executed hardware logic on the K-mer graph or nucleic acid sequences of a pileup associated with the K-mer graph identifier.

Claim 25. The method of claim 15, the method further comprising:

evaluating the K-mer graph to check for the existence of graph cycles;

if a graph cycle is detected during the evaluation:

terminating generation of the k-mer graph; or

if a graph cycle is not detected during the evaluation:

obtaining data from the hash table cache describing the structure of the k-mer graph; and

providing, to a variant calling module, the obtained data that describes the structure of the k-mer graph.

Claim 26. A system for hardware-accelerated generation of a K-mer graph using a programmable logic device, the system comprising:

a hardware-accelerated graph generation unit that includes hardware digital logic circuits that have been arranged to perform operations, the operations comprising:

obtaining a first set of nucleic acid sequences, wherein the first set of nucleic acid sequences include (i) a plurality of reads corresponding to a region of a reference sequence and (ii) a portion of the reference sequence;

for each particular nucleic acid sequence of the first set of nucleic acid sequences:

generating, for storage in a hash table cache and by a first hardware logic unit, data representing a graph node for each K-mer of the particular nucleic acid sequence;

detecting, by a control machine, that the first hardware logic unit has completed generation of a graph node for each K-mer of the particular nucleic acid sequence;

configuring, by the control machine, a second hardware logic unit to perform graph edge generation for the generated graph nodes; and

for one or more pairs of the generated graph nodes:

generating, by the second hardware logic unit and for storage in the graph hash table, data representing graph edges between one or more pairs of the generated graph nodes generated by the first hardware logic unit, wherein the data representing the graph node for each K-mer stored in the hash table cache and the data representing graph edges stored in the hash table cache represents a K-mer graph of the first set of nucleic acid sequences.

Claim 27. The system of claim 26, the operations further comprising:

periodically storing, by the control machine and in a memory unit that is accessible by the control machine, graph description data for an instance of the K-mer graph, wherein the graph description data represents (i) a K-mer graph identifier and (ii) K-mer graph state information.

Claim 28. The system of claim 26,

wherein the first hardware logic unit is further configured to:

determine whether one or more of the particular K-mers of the particular nucleic acid sequence matches another K-mer of the particular nucleic acid sequence; and

based on a determination that the one or more particular K-mers of the particular nucleic acid sequence match another K-mer of the particular nucleic acid sequence, store data that marks the one or more particular K-mers as non-unique K-mers.

Claim 29. The system of claim 26, wherein the second hardware logic is further configured to:

assign an edge weight to each edge of the K-mer graph.

Claim 30. The system of claim 26, the operations further comprising:

instructing a third hardware logic unit of the programmable logic device to execute hardware logic configured to:

obtain data representing the K-mer graph from the hash table cache; and

provide the obtained data representing the K-mer graph to a variant calling unit.

Claim 31. The system of claim 26, the operations further comprising:

instructing a third hardware logic unit of the programmable logic device to execute hardware logic configured to:

selectively delete data representing graph nodes and data representing graph edges of the K-mer graph from the hash table cache.

Claim 32. The system of claim 26, wherein the hash table cache is implemented using a third hardware logic unit of the programmable logic device.

Claim 33. The system of claim 26, wherein the graph description data further include (iii) data representing a last hardware logic unit of the plurality of hardware logic units that executed hardware logic on the K-mer graph or nucleic acid sequences of a pileup associated with the K-mer graph identifier.

Claim 34. The system of claim 26, the operations further comprising:

evaluating the K-mer graph to check for the existence of graph cycles;

if a graph cycle is detected during the evaluation:

terminating generation of the k-mer graph; or

if a graph cycle is not detected during the evaluation:

obtaining data from the hash table cache describing the structure of the k-mer graph; and

providing, to a variant calling module, the obtained data that describes the structure of the k-mer graph.

Claim 35. A hardware-accelerated graph generation unit that includes hardware digital logic circuits that have been arranged to perform operations, the operations comprising:

obtaining a first set of nucleic acid sequences, wherein the first set of nucleic acid sequences includes (i) a plurality of reads corresponding to a region of a reference sequence and (ii) a portion of the reference sequence;

generating, using a plurality of non-pipelined hardware logic units of a programmable logic device a K-mer graph using the obtained first set of nucleic acid sequences, wherein each hardware logic unit comprises a different hardware logic circuit configured to perform one or more operations, wherein each node of the K-mer graph represents a K-mer, each edge of the K-mer graph represents a link between a pair of K-mers, and each weight of each edge of the K-mer graph represents a number of occurrences of a K-mer sequence represented by a pair of K-mers; and

during generation of the K-mer graph:

periodically updating, with a control machine, graph description data for the K-mer graph after performance of the one or more operations by each hardware logic unit that is used to generate at least a portion of the K-mer graph, wherein the graph description data represents (i) a K-mer graph identifier and (ii) K-mer graph state information, wherein the control machine creates a workflow of operations using the non-pipelined hardware logic units by triggering performance of the one or more operations of each respective hardware logic unit during generation of the K-mer graph.

Claim 36. The hardware-accelerated graph generation unit of claim 35, wherein the output of each hardware logic unit of the plurality of hardware logic units is stored via a hash table cache.

Claim 37. The hardware-accelerated graph generation unit of claim 35, wherein the operations further include:

providing the generated K-mer graph to a variant calling unit, wherein the variant calling unit is configured to processs the K-mer graph to determine candidate variants between one or more of the plurality of reads and the reference sequence.

Claim 38. The hardware-accelerated graph generation unit of claim 35, wherein the operations further include:

obtaining, by a variant calling unit, the generated K-mer graph; and

identifying, based on the variant calling unit processing the generated K-mer graph, one or more candidate variants, wherein a candidate variant is a difference between a base call of one or more reads in a pileup of reads and a nucleotide of a reference genome at a particular location of the reference genome.

Claim 39. The hardware-accelerated graph generation unit of claim 35, wherein the graph description data further includes (iii) data representing a last hardware logic unit of the plurality of hardware logic units that executed hardware logic on the K-mer graph or nucleic acid sequences of a pileup associated with the K-mer graph identifier.

Claim 40. A hardware-accelerated graph generation unit that includes hardware digital logic circuits that have been arranged to perform operations, the operations comprising:

obtaining a first set of nucleic acid sequences, wherein the first set of nucleic acid sequences include (i) a plurality of reads corresponding to a region of a reference sequence and (ii) a portion of the reference sequence;

for each particular nucleic acid sequence of the first set of nucleic acid sequences:

generating, for storage in a hash table cache and by a first hardware logic unit, data representing a graph node for each K-mer of the particular nucleic acid sequence;

detecting, by a control machine, that the first hardware logic unit has completed generation of a graph node for each K-mer of the particular nucleic acid sequence;

configuring, by the control machine, a second hardware logic unit to perform graph edge generation for the generated graph nodes; and

for one or more pairs of the generated graph nodes:

generating, by the second hardware logic unit and for storage in the graph hash table, data representing graph edges between one or more pairs of the generated graph nodes generated by the first hardware logic unit, wherein the data representing the graph node for each K-mer stored in the hash table cache and the data representing graph edges

stored in the hash table cache represents a K-mer graph of the first set of nucleic acid sequences.

Claim 41. The hardware-accelerated graph generation unit of claim 40, the operations further comprising:

periodically storing, by the control machine and in a memory unit that is accessible by the control machine, graph description data for an instance of the K-mer graph, wherein the graph description data represents (i) a K-mer graph identifier and (ii) K-mer graph state information.

Claim 42. The hardware-accelerated graph generation unit of claim 40,

wherein the first hardware logic unit is further configured to:

determine whether one or more of the particular K-mers of the particular nucleic acid sequence matches another K-mer of the particular nucleic acid sequence; and

based on a determination that the one or more particular K-mers of the particular nucleic acid sequence match another K-mer of the particular nucleic acid sequence, store data that marks the one or more particular K-mers as non-unique K-mers.

Claim 43. The hardware-accelerated graph generation unit of claim 40, wherein the second hardware logic is further configured to:

assign an edge weight to each edge of the K-mer graph.

Claim 44. The hardware-accelerated graph generation unit of claim 40, the operations further comprising:

instructing a third hardware logic unit of the programmable logic device to execute hardware logic configured to:

obtain data representing the K-mer graph from the hash table cache; and

provide the obtained data representing the K-mer graph to a variant calling unit.

Claim 45. The hardware-accelerated graph generation unit of claim 40, the operations further comprising:

instructing a third hardware logic unit of the programmable logic device to execute hardware logic configured to:

selectively delete data representing graph nodes and data representing graph edges of the K-mer graph from the hash table cache.

Claim 46. The hardware-accelerated graph generation unit of claim 40, wherein the hash table cache is implemented using a third hardware logic unit of the programmable logic device.

Claim 47. The hardware-accelerated graph generation unit of claim 40, wherein the graph description data further include (iii) data representing a last hardware logic unit of the plurality of hardware logic units that executed hardware logic on the K-mer graph or nucleic acid sequences of a pileup associated with the K-mer graph identifier.

Claim 48. The hardware-accelerated graph generation unit of claim 40, the operations further comprising:

evaluating the K-mer graph to check for the existence of graph cycles;

if a graph cycle is detected during the evaluation:

terminating generation of the k-mer graph; or

if a graph cycle is not detected during the evaluation:

obtaining data from the hash table cache describing the structure of the k-mer graph; and

providing, to a variant calling module, the obtained data that describes the structure of the k-mer graph.