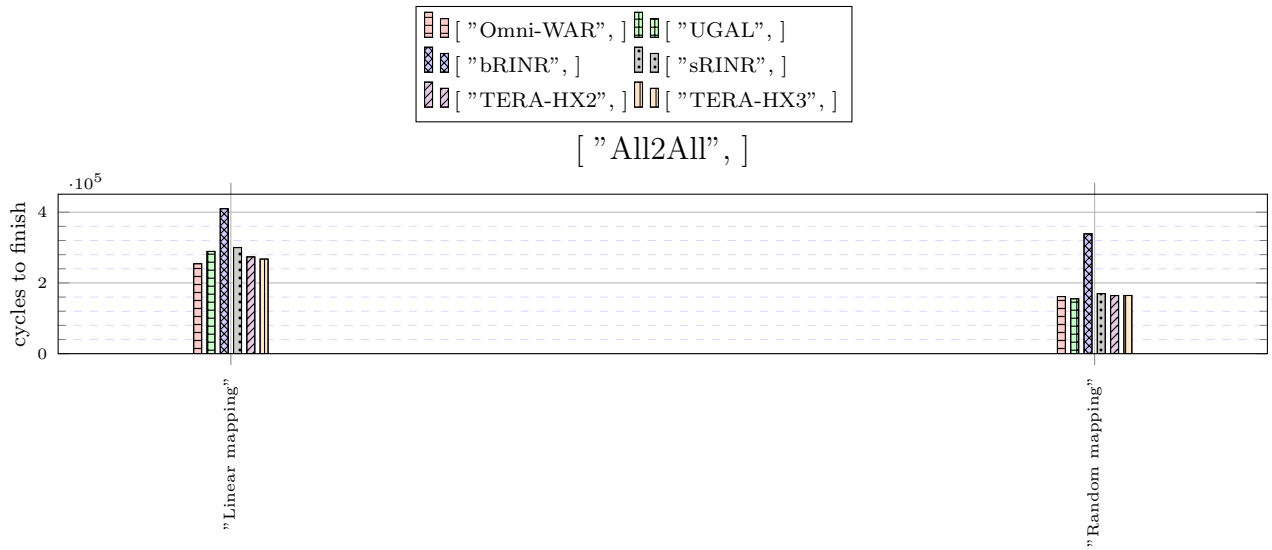

Figure 1: X: ["All2All",]

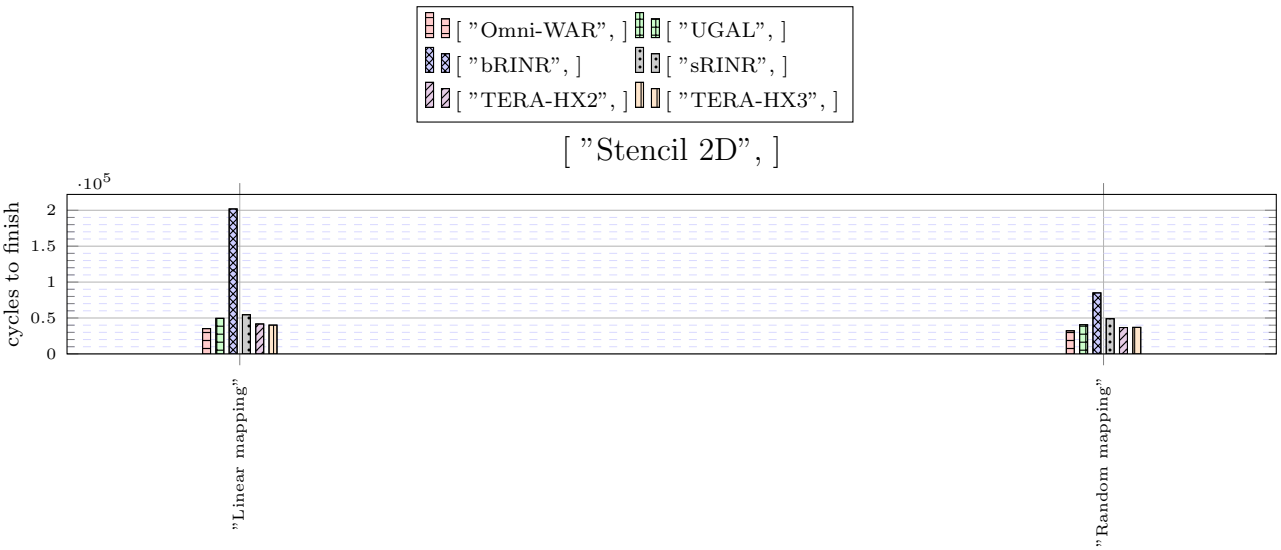

Figure 2: X: ["Stencil 2D",]

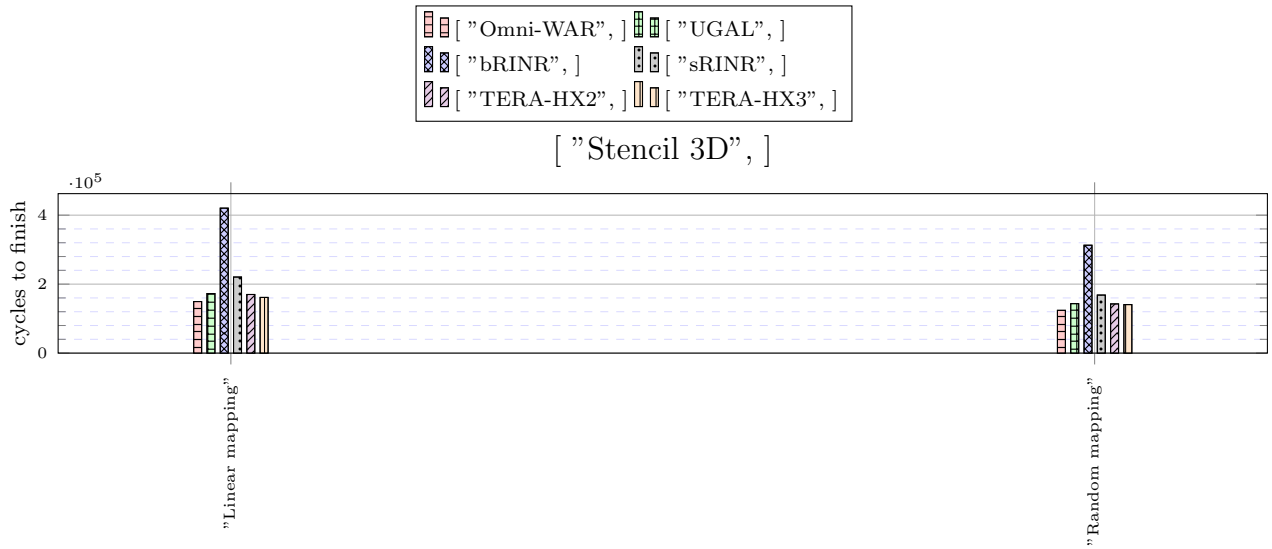

Figure 3: X: ["Stencil 3D",]

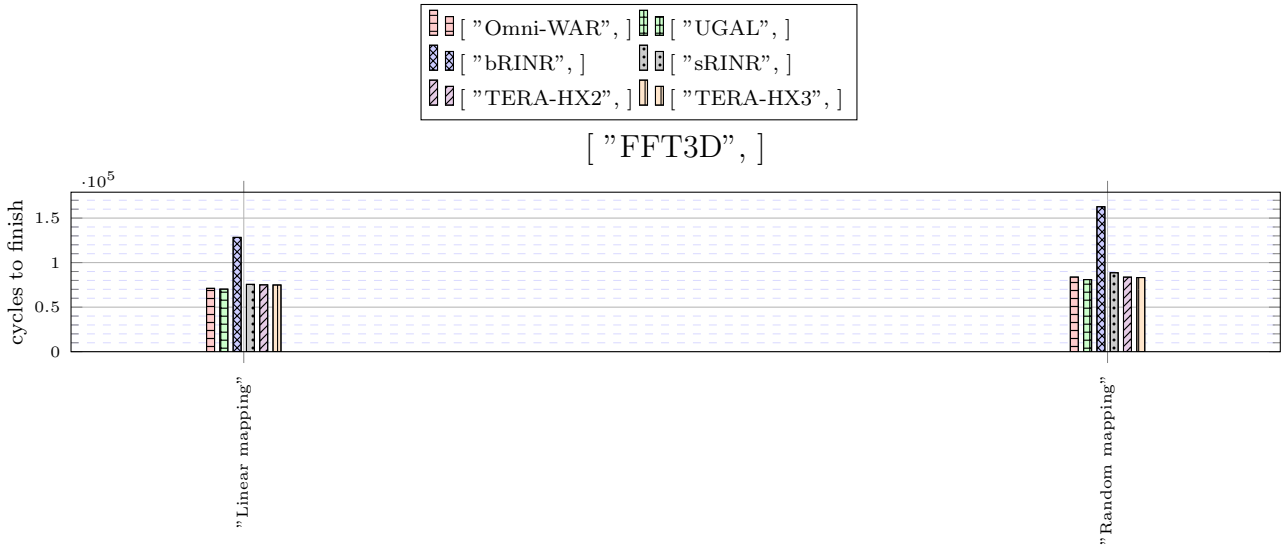

Figure 4: X: ["FFT3D",]

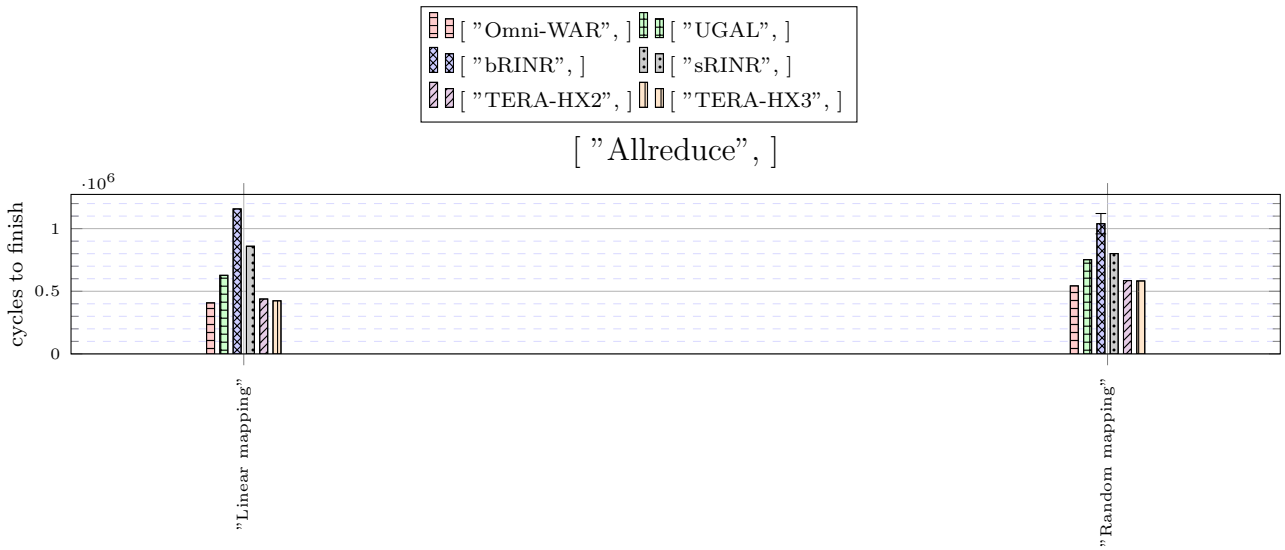

Figure 5: X: ["Allreduce",]

The following versions used in the simulations.

- heads/alex-stable-release-TERA-bc545f9093be36b0a173d3eb574ed35e54e2c29c(0.6.3)
- heads/alex-stable-00bacb613dc89a579028965f2bb90ce93ecd6e7a(0.6.3)
- heads/alex-stable-release-TERA-35e794829433712d102999df79a051a0256a860a(0.6.3)


```
Configuration{
  random_seed: ![ 1, 2, 3 ],
  warmup: 10000000000000,
  measured: 100000000000000,
  statistics_server_percentiles: [ 0, 5, 25, 50, 75, 95, 100 ],
  statistics_packet_percentiles: [ 0, 5, 25, 50, 75, 95, 100 ],
  general_frequency_divisor: 2,
  statistics_temporal_step: 1000,
  statistics_packet_definitions: [
    [
      [ ],
      [ ]]],
  topology: Hamming{
    servers_per_router: 64,
    sides: [ 64 ]},
  traffic: TrafficMap{
    tasks: 4096,
    map: ![
      Identity{ legend_name: "Linear mapping" },
      RandomPermutation{ legend_name: "Random mapping" }],
    application: ![
      All2All{ tasks: 4096, data_size: 65536, legend_name: "All2All" },
      Stencil{
        tasks: 4096,
        one_to_many_pattern: KingNeighbours{
          sides: [ 64, 64 ],
          distance: 1},
        message_size: 1024,
        rounds: 1,
        legend_name: "Stencil 2D"},
      Stencil{
        tasks: 4096,
        one_to_many_pattern: KingNeighbours{
          sides: [ 16, 16, 16 ],
          distance: 1},
        message_size: 1024,
        rounds: 1,
        legend_name: "Stencil 3D"},
      All2AllLinear{
        task_space: [ 64, 64 ],
        message_size: 256,
        rounds: 16,
        legend_name: "FFT3D"},
      AllReduce{ tasks: 4096, data_size: 65536, algorithm: Hypercube, legend_name: "Allreduce" }]},
  router: InputOutput{
    virtual_channels: mecanismo![ 2, 2, 1, 1, 1, 1 ],
    virtual_channel_policies: mecanismo![
      [
        WideHops{ width: 1 },
        VecLabel{
          label_vector: [ 0, 64 ]},
        OccupancyFunction{ label_coefficient: 1, occupancy_coefficient: 1, product_coefficient: 0, constant_coefficient: 0, use_internal_space: true,
use_neighbour_space: true, aggregate: true },
        OccupancyFunction{ label_coefficient: 1, occupancy_coefficient: 1, product_coefficient: 0, constant_coefficient: 0, use_internal_space: true,
use_neighbour_space: true, aggregate: false },
        LowestLabel,
        EnforceFlowControl,
        Random],
      [
        VecLabel{
          label_vector: [ 0, 64 ]},
        OccupancyFunction{ label_coefficient: 1, occupancy_coefficient: 1, product_coefficient: 0, constant_coefficient: 0, use_internal_space: true,
use_neighbour_space: true, aggregate: true },
        OccupancyFunction{ label_coefficient: 1, occupancy_coefficient: 1, product_coefficient: 0, constant_coefficient: 0, use_internal_space: true,
use_neighbour_space: true, aggregate: false },
        LowestLabel,
        EnforceFlowControl,
        Random],
      [
        VecLabel{
          label_vector: [ 0, 56 ]},
        OccupancyFunction{ label_coefficient: 1, occupancy_coefficient: 1, product_coefficient: 0, constant_coefficient: 0, use_internal_space: true,
use_neighbour_space: true, aggregate: true },
        LowestLabel,
        EnforceFlowControl,
        Random],
      [
        VecLabel{
          label_vector: [ 0, 56 ]},
        OccupancyFunction{ label_coefficient: 1, occupancy_coefficient: 1, product_coefficient: 0, constant_coefficient: 0, use_internal_space: true,
use_neighbour_space: true, aggregate: true },
        LowestLabel,
        EnforceFlowControl,
        Random],
      [
        VecLabel{
          label_vector: [ 0, 56 ]},
        OccupancyFunction{ label_coefficient: 1, occupancy_coefficient: 1, product_coefficient: 0, constant_coefficient: 0, use_internal_space: true,
use_neighbour_space: true, aggregate: true },
        LowestLabel,
        EnforceFlowControl,
        Random],
      [
        VecLabel{
          label_vector: [ 0, 56, 56 ]},
        OccupancyFunction{ label_coefficient: 1, occupancy_coefficient: 1, product_coefficient: 0, constant_coefficient: 0, use_internal_space: true,
```

```
use_neighbour_space: true, aggregate: true },
        LowestLabel,
        EnforceFlowControl,
        Random]],
    allocator: Random,
    buffer_size: 160,
    bubble: false,
    flit_size: 16,
    intransit_priority: false,
    allow_request_busy_port: true,
    output_buffer_size: 80,
    crossbar_frequency_divisor: 1,
    crossbar_delay: 2},
  maximum_packet_size: 16,
  routing: mecanismo![
    OmniDimensionalDeroute{ allowed_deroutes: 1, include_labels: true, legend_name: "Omni-WAR" },
    Sum{
      policy: TryBoth,
      first_routing: Shortest,
      second_routing: Valiant{
        first: Shortest,
        second: Shortest,
        first_reserved_virtual_channels: [ 0 ],
        second_reserved_virtual_channels: [ 1 ]},
      first_allowed_virtual_channels: [ 0 ],
      second_allowed_virtual_channels: [ 0, 1 ],
      second_extra_label: 1,
      legend_name: "UGAL"},
    Sum{
      policy: TryBoth,
      first_routing: Shortest,
      second_routing: CGLabel{ balance_algorithm: bRINR },
      first_allowed_virtual_channels: [ 0 ],
      second_allowed_virtual_channels: [ 0 ],
      second_extra_label: 1,
      legend_name: "bRINR"},
    Sum{
      policy: TryBoth,
      first_routing: Shortest,
      second_routing: CGLabel{
        balance_algorithm: sRINR{ a: 0, b: 0 },
        weight_repetition: true},
      first_allowed_virtual_channels: [ 0 ],
      second_allowed_virtual_channels: [ 0 ],
      second_extra_label: 1,
      legend_name: "sRINR"},
    SubTopologyRouting{
      logical_topology: Hamming{
        servers_per_router: 2,
        sides: [ 8, 8 ]},
      map: Identity,
      logical_routing: DOR{
        order: [ 0, 1 ]},
      opportunistic_hops: true,
      legend_name: "TERA-HX2"},
    SubTopologyRouting{
      logical_topology: Hamming{
        servers_per_router: 2,
        sides: [ 4, 4, 4 ]},
      map: Identity,
      logical_routing: DOR{
        order: [ 0, 1, 2 ]},
      opportunistic_hops: true,
      livelock_avoidance: true,
      legend_name: "TERA-HX3"}],
  link_classes: [
    LinkClass{ delay: 2 },
    LinkClass{ delay: 2 },
    LinkClass{ delay: 2 },
    LinkClass{ delay: 2 }],
  launch_configurations: [
    Slurm{ job_pack_size: 1, time: "2-10:00:00" }]}
```