
Algorithm 1 delta-stepping GPU APSP

Input: $G(V, E)$, source vertex s ;

Output: $dist(v)$, ($v \in V$), the weight of the shortest path from s to v ;

```
1:
2: function initial( $V$ )
3:   for each  $u \in V$  do
4:     for each  $v \in V$  do
5:        $dist(u)(v) \leftarrow +\infty$ ; ▷ initialize  $dist$  to positive infinity;
6:     end for
7:   end for
8:    $dist(u)(u) \leftarrow 0$ ; ▷ set the source distance to 0;
9: end function
10:
11: function delta_steppingCudaFunc( $G(V, E)$ ,  $dist$ ,  $predist$ ) ▷  $G(V, E)$ , the initially distance array
     $dist$ , a temporary distance array  $predist$ , the delta  $\Delta$ ;
12:    $u0 \leftarrow threadId$ ; ▷ get the thread id;
13:    $offset \leftarrow blockDim$ ; ▷ get the number of threads in a block;
14:    $s0 \leftarrow blockIdx$ ; ▷ get the block id;
15:    $blockNum \leftarrow gridDim$ ; ▷ get the number of blocks in all grids;
16:    $id \leftarrow (\_shared\_memory) 1$ ; ▷ the bucket id;
17:    $B \leftarrow (\_shared\_memory)\emptyset$ ; ▷ set the bucket  $B$  to emptyset;
18:
19:    $s \leftarrow s0$ ; ▷ set the source vertex in a block;
20:   while  $s < |V|$  do
21:     while  $B \neq \text{emptyset}$  do
22:        $Req \leftarrow \emptyset$ ; ▷ the vertices used to be in current bucket;
23:       while  $B(id) \neq \text{emptyset}$  do
24:          $u = u0$ ;
25:         while  $u < |V|$  do
26:           if  $u \in B(id)$  then
27:              $Req \leftarrow Req \cup \{u\}$ ;
28:             for each  $(u, v, w) \in E$  do
29:               if  $w \leq \Delta$  then ▷ the light edge;
30:                  $atomicMin(\&predist(s)(v), dist(s)(u) + w)$ ;
31:               end if
32:             end for
33:           end if
34:            $u \leftarrow (u + offset)$ ;
35:         end while
36:
37:        $\_syncthreads()$ ;
38:        $u \leftarrow u0$ ;
39:       while  $u < |V|$  do
40:         if  $predist(s)(u) < dist(s)(u)$  then
41:            $dist(s)(u) = predist(s)(u)$ ;
42:            $moveToB(dist(s)(u)/\Delta)$ 
43:         end if
44:          $u \leftarrow (u + offset)$ ;
45:       end while
```

```

46:      end while
47:
48:      --syncthreads();
49:      u = u0;
50:      while u < |V| do
51:          if u ∈ Req then
52:              for each  $(u, v, w) \in E$  do
53:                  if  $w > \Delta$  then ▷ the heavy edge;
54:                      atomicMin(&predist(s)(v), dist(s)(u) + w);
55:                  end if
56:              end for
57:          end if
58:          u ← (u + offset);
59:      end while
60:
61:      --syncthreads();
62:      u ← u0;
63:      while u < |V| do
64:          if  $\text{predist}(s)(u) < \text{dist}(s)(u)$  then
65:               $\text{dist}(s)(u) = \text{predist}(s)(u)$ ;
66:              moveutoB(dist(s)(u)/Δ)
67:          end if
68:          u ← (u + offset);
69:      end while
70:
71:      id ← (id + 1); ▷ goto next bucket;
72:      end while
73:      s ← (s + blockNum)
74:  end while
75: end function
76:
77: initial(V);
78:
79: host_to_device(dist), host_to_device(G(V, E)); ▷ copy the dist and
   G(V, E) from main memory to GPU memory;
80:
81: delta_steppingCudaFunc(); ▷ call the CUDA kernal;
82: device_to_host(dist); ▷ copy the dist back;
83:
84: return result

```
