Solving Consensus using S (worst case scenario)

5 processes: P1, P2, P3, P4, P5
No process crashes: strong completeness is (trivially) satisfied
P5 is never suspected: eventual weak accuracy is satisfied

Initially:

```
V1 = v1  ⊥  ⊥  ⊥  ⊥        Δ1 = v1  ⊥  ⊥  ⊥  ⊥
V2 =  ⊥ v2  ⊥  ⊥  ⊥        Δ2 =  ⊥ v2  ⊥  ⊥  ⊥
V3 =  ⊥  ⊥ v3  ⊥  ⊥        Δ3 =  ⊥  ⊥ v3  ⊥  ⊥
V4 =  ⊥  ⊥  ⊥ v4  ⊥        Δ4 =  ⊥  ⊥  ⊥ v4  ⊥
V5 =  ⊥  ⊥  ⊥  ⊥ v5        Δ5 =  ⊥  ⊥  ⊥  ⊥ v5
```

r = 1
P1-P2-P3-P4 receive Δ5 from P5: they all learn v5
P5 receives Δ4 from P4: it learns v4
P4 receives Δ3 from P3: it learns v3
P3 receives Δ2 from P2: it learns v2
P2 receives Δ1 from P1: it learns v1

```
V1 = v1  ⊥  ⊥  ⊥ v5        Δ1 =  ⊥  ⊥  ⊥  ⊥ v5
V2 = v1 v2  ⊥  ⊥ v5        Δ2 = v1  ⊥  ⊥  ⊥ v5
V3 =  ⊥ v2 v3  ⊥ v5        Δ3 =  ⊥ v2  ⊥  ⊥ v5
V4 =  ⊥  ⊥ v3 v4 v5        Δ4 =  ⊥  ⊥ v3  ⊥ v5
V5 =  ⊥  ⊥  ⊥ v4 v5        Δ5 =  ⊥  ⊥  ⊥ v4  ⊥
```

P5 has some information (v4) that P1-P2-P3 do not have
If we go directly to Phase 2, P1-P2-P3 will decide v5, but P4-P5 will decide v4

r = 2
P1-P2-P3-P4 receive Δ5 from P5: P1-P2-P3 learn v4
P5 receives Δ4 from P4: it learns v3
P4 receives Δ3 from P3: it learns v2
P3 receives Δ2 from P2: it learns v1

```
V1 = v1  ⊥  ⊥ v4 v5        Δ1 =  ⊥  ⊥  ⊥ v4  ⊥
V2 = v1 v2  ⊥ v4 v5        Δ2 =  ⊥  ⊥  ⊥ v4  ⊥
V3 = v1 v2 v3 v4 v5        Δ3 = v1  ⊥  ⊥ v4  ⊥
V4 =  ⊥ v2 v3 v4 v5        Δ4 =  ⊥ v2  ⊥  ⊥  ⊥
V5 =  ⊥  ⊥ v3 v4 v5        Δ5 =  ⊥  ⊥ v3  ⊥  ⊥
```

P5 has some information (v3) that P1-P2 do not have
If we go directly to Phase 2, P1-P2 will decide v4, but P3-P4-P5 will decide v3

Mikel Larrea, Departamento de Arquitectura y Tecnología de Computadores, UPV/EHU

r = 3
P1-P2-P3-P4 receive Δ5 from P5: P1-P2 learn v3
P5 receives Δ4 from P4: it learns v2
P4 receives Δ3 from P3: it learns v1

```
V1 = v1  ⊥ v3 v4 v5        Δ1 =  ⊥  ⊥ v3  ⊥  ⊥
V2 = v1 v2 v3 v4 v5        Δ2 =  ⊥  ⊥ v3  ⊥  ⊥
V3 = v1 v2 v3 v4 v5        Δ3 =  ⊥  ⊥  ⊥  ⊥  ⊥
V4 = v1 v2 v3 v4 v5        Δ4 = v1  ⊥  ⊥  ⊥  ⊥
V5 =  ⊥ v2 v3 v4 v5        Δ5 =  ⊥ v2  ⊥  ⊥  ⊥
```

P5 has some information (v2) that P1 does not have
If we go directly to Phase 2, P1 will decide v3, but P2-P3-P4-P5 will decide v2

r = 4
P1-P2-P3-P4 receive Δ5 from P5: P1 learns v2
If P5 receives Δ4 from P4, it learns v1, else it does not learn anything

```
V1 = v1 v2 v3 v4 v5        Δ1 =  ⊥ v2  ⊥  ⊥  ⊥
V2 = v1 v2 v3 v4 v5        Δ2 =  ⊥  ⊥  ⊥  ⊥  ⊥
V3 = v1 v2 v3 v4 v5        Δ3 =  ⊥  ⊥  ⊥  ⊥  ⊥
V4 = v1 v2 v3 v4 v5        Δ4 =  ⊥  ⊥  ⊥  ⊥  ⊥
V5 =  ⊥ v2 v3 v4 v5        Δ5 =  ⊥  ⊥  ⊥  ⊥  ⊥
```

Now all processes have as much information as P5, so we can go to Phase 2

In Phase 2, processes exchange their V vectors. As they will at least receive
V5 from P5, P1-P2-P3-P4 will 'forget' the first component of their vectors

```
V1 =  ⊥ v2 v3 v4 v5
V2 =  ⊥ v2 v3 v4 v5
V3 =  ⊥ v2 v3 v4 v5
V4 =  ⊥ v2 v3 v4 v5
V5 =  ⊥ v2 v3 v4 v5
```

In Phase 3, the unanimous decision will be v2

(Note: if P5 received Δ4 from P4 in round 4, the decision would be v1)