



Gres = {Vres e , Vres c , Eres}

Let V = {Vres c , Vres e } such that |V | = L.

Vres e represents the set of edge nodes, Vres c represents the set of cloud nodes, and Eres represents the set of logical links between the nodes

Gtop = {So, Op, Si, Etop}

T = {So ∪ Op ∪ Si} and |T | = K

So represents the set of sources, Op represents the set of operators, Si represents the set of sinks, and Etop represents the set of streaming data flows between the vertices in the graph.

a

In a typical DSP scenario, the sinks, sources and operators, known as tasks, are placed and executed on a processing node

For a given operator Tj:

selectivity ratio σT → throughput



productivity ratio ρTj, → message size





A Constraint Satisfaction Problem (CSP) can be defined as θ = (α, β, C)

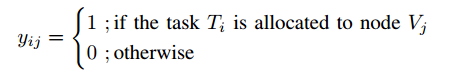
A : T → V

Ti → A(Ti) = νj.

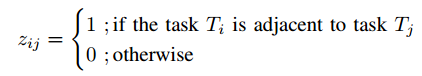
find a mapping A∗ ∈ A such that ∀A, F(A∗) ≤ F(A).



K × L boolean variables



K × K boolean variables





some tasks may be required to be placed, or pinned, on specific resources.

We can assume that the sources of the topology need to be placed on the edge devices.

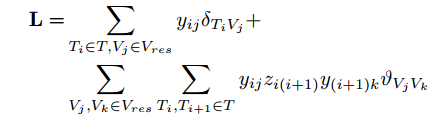


We look at several resources in this regard, CPU, bandwidth, and energy



minimise the end-to-end latency of the DSP topology as it runs on the underlying network, since edge-cloud applications are known to suffer from high end-to-end latency

the end-toend latency is the time between an input tuple being generated at a source node to when the results concerning that input tuple arrive at the sink node.



where δTiVj is the processing time of task Ti at the host Vj, and ϑVjVk is the latency of transferring an event between two hosts.