**Problem**
- We schedule a set of jobs.
- We can switch-on more servers.
- We can vary the speed of processors.

Our goal is to minimize energy consumption.

**Resolution**
We build our scheduling in three phases.
- First, we obtain a workload estimation of the next period by adjusting a predictor.
- Next, we determine the optimal number of server to minimize energy consumption for this estimated workload.
- Finally, we dispatch the estimated number of jobs on the switched-on servers.

**Experimentation**
We compare our algorithms to an off-line optimal algorithm. The three algorithms we use is:
- Naive which use the previous workload to choose the number of servers.
- Non dynamic which use a non-dynamic bandit for the prediction.
- Dynamic which use a dynamic bandit.

Choose the number of servers
We prove that the energy consumption decreases until a certain number of switched-on servers. Then energy increases if we add more servers. So we can find the optimal number of server with a dichotomic search.

Dispatching
Once we have choose our optimal number of servers, we evenly dispatch our jobs on the processors and we run them as slowly as possible.

Prediction
A non-dynamic bandit is:

For a dynamic bandit: at each time $t$, we compute $A_{opt}(t)$, the value of an optimal arm. When we have $m$ arms, we evenly choose $n$ arms between these $m$ arms.

Here is an illustration of this construction for $n = 11$ and $m = 55$. The selected arms are shadowed:

At time $t$, our bandit configuration is composed of the arms in red:

**Results**

![Energy consumption and number of incoming jobs](image)

Almost perfect prediction, $N(0,3\%)$

Slightly overestimated prediction, $N(5\%, 3\%)$

Strongly underestimated prediction, $N(-15\%, 3\%)$