Energy aware scheduling

March 9, 2016

1 Problem formulation

We consider a problem of applications scheduling in a center with limited power and resources.

This data center runs two different applications : active applications (also called web applications) and batch applications.

1.1 Active applications

We consider multiple active applications $\{A_i\}, i \in 1..n$, which run continuously over a given amount of time intervals 1..24.

Each application A_i has, at a given time interval j, an execution modes ; each application's mode produces a given power consumption and profit when applied over an interval, with

$M_{i,j} \in 1..3$ mode of activity i at time j.

$E_{i,j} \in 0..1000$ power consumption of activity i at time j.

$P_{i,j} \in 0..1000$ profit for running activity i at time j.

Those informations can be stored in tables, eg :

	Mode 1	Mode 2	Mode 3
A_1	50	60	70
A_2	40	45	55

Table 1: Energy consumption for two active applications

	Mode 1	Mode 2	Mode 3
A_1	100	110	120
A_2	90	100	110

Table 2: Profit for two active applications

- element ($M_{i,j}$, Row_i^{Energy}, $E_{i,j}$) % E_i=$Row_i^{Energy}[M_i]$
- element ($M_{i,j}$, Row_i^{Profit}, $P_{i,j}$) % P_i=$Row_i^{Profit}[M_i]$

1.2 Batch applications

We consider multiple batch jobs $\{B_i\}, i \in 1..m$, each with its own parameters :

$Duration_i$ is the number of intervals this job must be run before being finished,

$Deadline_i$ deadline is the interval at which the job must be finished,

Q_i $\in$ 0..1000 is the profit earnt only if the whole job is finished before or at deadline,

$Power_i$ is the power consumed by the job when it is run over an interval

The batch jobs do not have execution modes.

	On	Off
B_1	60	0
B_2	80	0

Table 3: Energy consumption for Batch applications

Each batch job i can be decomposed in as many subjobs i, j as its duration. We note $S_{i,j}$ the interval at which the j^{th} subjob of jow i is executed. Since each subjob must be executed over a different time interval,
$0 < S_{i,1} < S_{i,2} ... < S_{i,Duration_i}$.
If the job meets its deadline, then also
$S_{i,Duration_i} <= Deadline_i$
Energy consumption of batch job i during an interval is $E_{i,j}$, where $i \in$ [1,m] and $j \in$ [1....$Duration_i$].

- element ($S_{i,Duration_i}$,[.........$Profit_i$, Pen_i, ...], Q_i)

1.3 Energy cap

We also introduce the maximum available energy at j as $Capacity_j$, where $j \in$ [1,24]. Since, in cumulative constraint modeling, the limit of maximum capacity of available energy can not be changed, we introduce fake jobs in each slot to match with the $Capacity_j$. So the maximum capacity is defined as Limit, where Limit = $max(Capacity_h)$, $h \in$ [1,24]. For scheduling purpose we slice the total time i.e., 24/48 hours to 24/48 slots meaning 1 hour as slot and schedule each slot in advance with known information. So, Total Profit is $P = \sum_{i=1}^{n} P_{i,j} + \sum_{i=1}^{m} Q_i$

- For each active job A_i starts at j, has duration of 1 slot and height of power consumption is E_{ij}, where, ∀j $\in$ [1,24]
- For each batch jobs B_i the start time is S_{ij}, duration is 1 slot and and height of power consumption is E_i, where, ∀i $\in$ [1,m] and ∀j $\in$ [1,$duration_i$]
- For every slot $j \in$ [1,24] we have a fake job F_h that starts at fixed j, duration of 1 slot and height of power consumption of *Limit* - $Capacity_j$, where, ∀j $\in$ [1,24]

1.4 Memory use

The data center contains l servers $\{S_k\}, k \in 1..l$. Each server S_k has a memory capacity M_k which limits the number of applications this server can execute.

Executing an application A_i or a job B_i on a server S_k consumes a fixed amount of memory on the server over the execution interval. Reciprocally, each application must be run on a server at any time and any job executed during an interval must be run on a server.

We note

$HA_{i,j}$ the host of the application A_i during interval j

$P(HA_{i,j})$ the cost of moving the application A_i from one host to $HA_{i,j}$. If $HA_{i,j-1}$ is defined i.e. application A_i was running during interval $j-1$, and $HA_{i,j-1} = HA_{i,j}$ then $P(HA_{i,j}) = 0$. If $HA_{i,j-1}$ is undefined, then $P(HA_{i,j}) = 0$. If $HA_{i,j-1}$ is defined and $HA_{i,j-1} \neq HA_{i,j}$ then $P(HA_{i,j})$ is a function of $HA_{i,j-1}$ and $HA_{i,j}$.

$HB_{i,j}$ the host of the job B_i during interval j

$Load_{k,j}$ the memory load of the server k during interval time j

$mem(C)$ the memory use of the job or application C.

then we know that

$$Load_{k,j} = \sum_{A_i} (mem(A_i) if HA_{i,j} = k) + \sum_{B_i} (mem(B_i) if HB_{i,j=k} \tag{1}$$

2 Solution formulation

A solution to such a problem consists in :

- $\forall$ Active application A_i and time interval j, the execution mode $M_{i,j}$ at which to run the appliction during the interval
- $\forall$ Batch job B_i, a series time interval $S_{i,1}..S_{i,Duration_i}$ at which to execute the job.
- $\forall$ time interval j, $\forall$ server S_k, the set of Applications and jobs hosted on the server during the time interval.

with respect to the previoulsy specified conditions.

3 Objective

Our objective function is to maximize P, over the 24 time slots
$P = \sum_{j=1}^{24} \left(\sum_{i=1}^{n} P_{i,j} + \sum_{i=1}^{m} Q_i + \sum_{i=1}^{n} P(HA_{i,j}) + \sum_{i=1}^{m} P(HB_{i,j}) \right)$