IEEE802.3poep Study Group

Classification Resolution Analysis

Preliminary Analysis – Part A

Darshan Yair / PowerDsine

San Francisco, June 2005

1. Scope

The purpose of the following analysis is to establish design rules for setting the classification resolution requirement that is needed to Improves PSE power supply utilization (PSu) hence reducing system cost, heat removal and improves reliability and system availability.

Power Management using power classification is based on the power class information given by the PD to the PSE during class detection phase prior to applying power to the PD.

The following analysis may be used as design criteria for any classification method and is not limited to the classification method used in the IEEE802.3af.

2. Objectives

2.1. What is the minimum class resolution?
2.2. How many classes do we need?
2.3. Well defined system design procedure

3. The Ideal Case

Theoretically the best solution would be that the power range addressed to be divided to the tiniest value in order to have the highest resolution hence the most efficient Power Management. However, this approach is neither cost effective nor practical for the following reasons:

- High resolution is not free. It increases silicon cost.

- There is an economical limit for the contribution of high classification resolution to the utilization of the PSE power supply (PSE PS). At some point, additional increase in resolution has a small effect on PSE PS utilization and the resolution marginal cost increases.

- There is a power level where power management has no effect on PSu.

4. The Practical Case

The simplest power management scheme and operating conditions for the purpose of illustrating the arguments presented in this document is shown below:

4.1 The PSE power supply has a power budget, Pbudget.
4.2 The PSE has N ports, each port support Pport which is a random variable.
4.3 Pport has an average value which is one of the parameters that specifies Pport probability curve.
4.4 In a known environment, each port has the same probability function, hence it generates the same average power per port.
4.5 The minimum power class resolution is Class_step.
4.6 Class step may be linear (constant value for any Pport value) or non linear (function of Pport). In the following analysis the linear case is discussed.
4.7 The objective is to optimize or maximize the PSu.

5. Detailed Analysis

PSE power supply utilization factor is defined as:

$$\boxed{PSu = \frac{Actual\_Power\_Taken}{Pbudget}} \qquad \text{Eq-1}$$

Through the whole analysis, we look for average results which will simplify the math work results for getting practical conclusions.

Averaging Eq-1 for getting PSu average:

$$\boxed{PSu\_avg = \frac{Actual\_Power\_Taken\_avg}{Pbudget\_avg}} \qquad \text{Eq-1.1}$$

Note: Pbudget is not a random variable due to the fact that it is function of the Actual_Power_Taken_avg as will be shown later. Therefore Pbudget_avg=Pbudget.

$$Actual\_Power\_Taken = \sum_{i=1}^{N} Pport_i \qquad \text{Eq-2}$$

Where:
-N is the number of ports
-Pport_i is the PSE port power of port number i, which is a random variable which is defined by its probability function. (Probability function assigns to each value of Pport_i out of m possible values the probability p_i for Pport=Pport_i.)
- In the general case, each port has its own probability function, $f_{ij}(Pport_j)$, where i is the port number and j is the index of the random variable Pport hence $p_{j=} f_{ij}(Pport_j)$.

In addition the following should be satisfied for and probability function: $\sum_{j=1}^{m} p_j = 1$ Eq-3.

j	$p_{j=} f_{ij}(Pport_j)$	$Pport_j$
1	0.3	5W
2	0.25	7W
3	0.15	11W
4	0.15	20W
5	0.13	25W
6	0.02	56W
	$\sum_{j=1}^{m} p_j = 1$	$Pport\_avg_i = \sum_{j=1}^{m} Pport_j \cdot f_{ij}(Pport) = 12.27W$

Table 1: Example of a probability function for a port in a given organization

Averaging equation 2 results with:

$$avg(Actual\_Power\_Taken) = avg\left(\sum_{i=1}^{N} Pport_i\right) \qquad \text{Eq-3.1}$$

$$Actual\_Power\_Taken\_avg = \sum_{i=1}^{N} Pport\_avg_i \qquad \text{Eq-4}$$

The average power for a port_i is:

$$\boxed{Pport\_avg_i = \sum_{j=1}^{m} Pport_j \cdot p_{ij} = \sum_{j=1}^{m} Pport_j \cdot f_{ij}(Pport)} \qquad \text{Eq-5}$$

Inserting Eq-4 IN Eq-5 results with:

$$\boxed{Actual\_Power\_Taken\_avg = \sum_{i=1}^{N} \sum_{i=1}^{m} Pload_j \cdot f_{ij}(Pload)} \qquad \text{Eq-6}$$

Eq-6 represents the general case of the actual average power taken from all PSE ports with any probability function for each port.

The probability function is mainly a function of the market being addressed and end user considerations. In this case the probability function can be anything.

In order to bypass this problem we can use condition 4.4 which can simplify the calculation of the average power taken by the ports.

Condition 4.4 states that in a known environment, it will be fair to assume that each port has the *same probability function*, hence it generates the *same average power per port*. See example in Table 1.

Hence: $f_{1j}(Pload) = f_{2j}(Pload) = ... f_{mj}(Pload)$ Eq-6.1

Therefore: $Pport\_avg_1 = Pport\_avg_2 = ...Pport\_avg_m = Pport\_avg$ Eq-6.2

Inserting the above reduces Eq-6 the following simple form:

$$Actual\_Power\_Taken\_avg =$$

$$\sum_{i=1}^{N} Pport\_avg_i = N \cdot Pport\_avg_i =$$ Eq-6.3

$$N \cdot Pport\_avg$$

$$Actual\_Power\_Taken\_avg = N \cdot Pport\_avg$$ Eq-8

If Eq-8 supply the data which is required for the actual average power taken then the budget, Pbudget, should be set to at least the same value in order to supply the demand. However, as explained in clause 1, the system reads the class information from the PD; the class information tells the PSE how much power to allocate for this port. The PD defines its class according to the actual maximum power that it needs and the minimum class step possible according to Eq-9.

The class is defined by the following equation:

$$\boxed{Pclass_i = \left(ceil\left(\frac{Pport_i}{Class\_step} \right) \right) \cdot Class\_step}$$ Eq-9

Where:
-Pclass is the class information in watts that describes Pport.
-Class step is the class resolution i.e. minimum difference between two consecutive classes.
-Ceil is a function that rounds up the number to the next integer.
If Port=4.5W and the minimum class step (Class resolution) is 2W than the actual class is 6W. (ceil(4.5/2))*2=6.

Averaging Eq-9 results with:

$$\boxed{Pclass\_avg_i = \sum_{j=1}^{m} Pclass_j \cdot f_{ij}(Pport)}$$ Eq-10

Inserting Eq-9 in Eq-10 results with:

$$\boxed{Class\_avg_i = \left(ceil\left(\frac{Pport\_avg_i}{Class\_step} \right) \right) \cdot Class\_step}$$ Eq-11

Hence the budget Pbudget should be set to support:

$$\boxed{\text{Pbudget} = Class\_avg_i \cdot N = Class\_avg \cdot N}$$ Eq-12

Inserting Eq-8 and Eq-11 in Eq-1:

$$\boxed{PSu\_avg = \frac{Actual\_Power\_Taken\_avg}{Pbudget} = \frac{Pport\_avg \cdot N}{Class\_avg \cdot N} = \frac{Pport\_avg}{Class\_avg}}$$ Eq-13

$$\boxed{PSu\_avg = \frac{Pport\_avg}{Class\_avg}}$$ Eq-14

If Class_step was small enough then Pclass_avg=Pport_avg.
So far, according to the above for any N, in average, PSu_avg is not a function of N.
In a working system Pclass>=Pport hence Pclass_avg>=Pport_avg.

Inserting Eq-11 in Eq-14:

$$PSu\_avg = \frac{Pport\_avg}{\left(ceil\left(\frac{Pport\_avg_i}{Class\_step}\right)\right) \cdot Class\_step}$$ Eq-15

Let's define the ratio between the average power per port, Pport_avg and the class_step as β:

$$\boxed{\beta = \frac{Pport\_avg}{Class\_step}}$$ Eq-16

β has a physical meaning of the average number of classes. Higher β is smaller class_step hence better resolution which results with higher number of classes.

Inserting Eq-16 in Eq-15

$$\boxed{PSu\_avg = \frac{\beta}{ceil(\beta)}}$$ Eq-17

$$PSu\_avg = \frac{\beta}{ceil(\beta)}$$ Eq-17

$$\beta = \frac{Pport\_avg}{Class\_step}$$ Eq-16

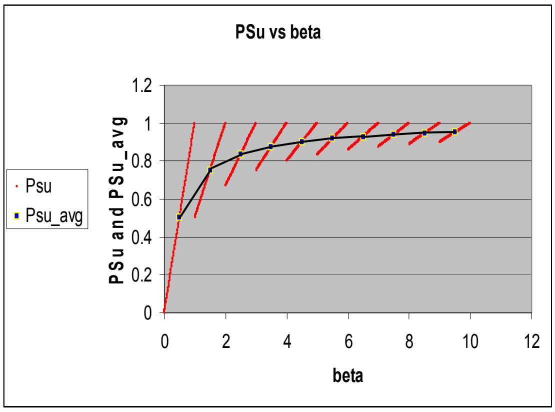

Figure 1: PSu and PSu avg as function of β

Conclusions:

PSu=100% for any integer of $\beta = \frac{Pport\_avg}{Class\_step}$ with low probability.

1.1 PSU<100% for any positive non integer β hence:

$$PSu\_avg < \frac{\beta}{ceil(\beta)} = \frac{Pport\_avg}{Class\_step \cdot ceil(\beta)}$$ Eq-18

1.2 The requirement for Class_step is : $Class\_step < \frac{Pport\_avg}{PSu\_avg \cdot ceil(\beta)}$ Eq-19

0 <β < βmax for the desired PSu_avg.

βmax is the β which $\frac{\partial PSu\_avg}{\partial \beta} < \varepsilon$ is no longer cost effective which is implementation dependent.

Example:

- For PSu_avg=90% βmin=4, βmax=5, βavg=4.5 ==> β=5 Changing β from 5 to 6 will change PSu_avg by only 2.47%-1.65%=0.82%.

In some class implementation, improving PSu_avg by 0.82% cost more than not utilizing additional 0.82% of the power supply capability and in some cost lees or nothing until we again cross a value of β that requires economical decision.

2. PSu utilization is mainly a function of the ratio between Pport_avg and the Class_Step
3. The rules applied for any port number and for any Pbudget in average as long as

$$Pbudget < \sum_{i=1}^{N} Pclass\_\max_i$$

Pclass_max is the class value of Pport_max which is the maximum possible port power. If otherwise, power management is not required due to the fact that PSE power supply supports all ports with max power.

6. Simulations results

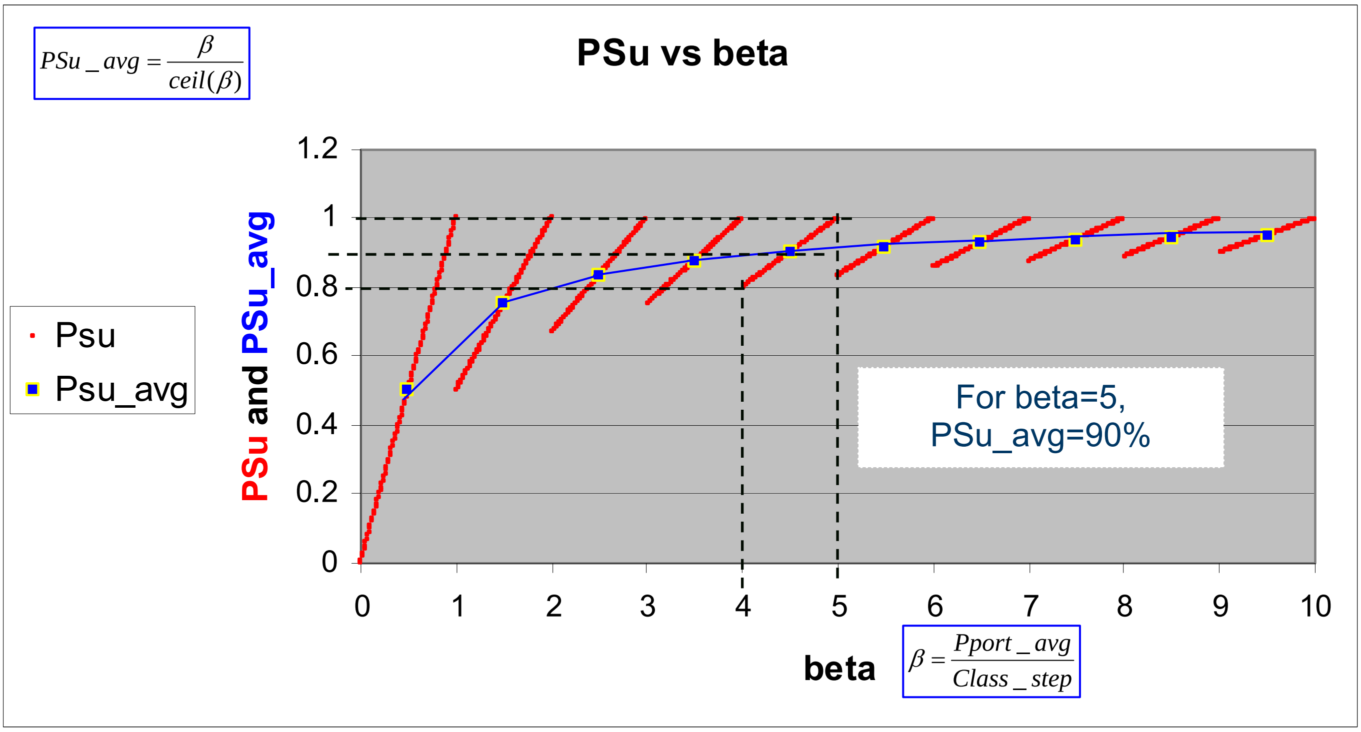

Figure 2: PSu_avg vs β. Example of PSu_avg=90% with β=5

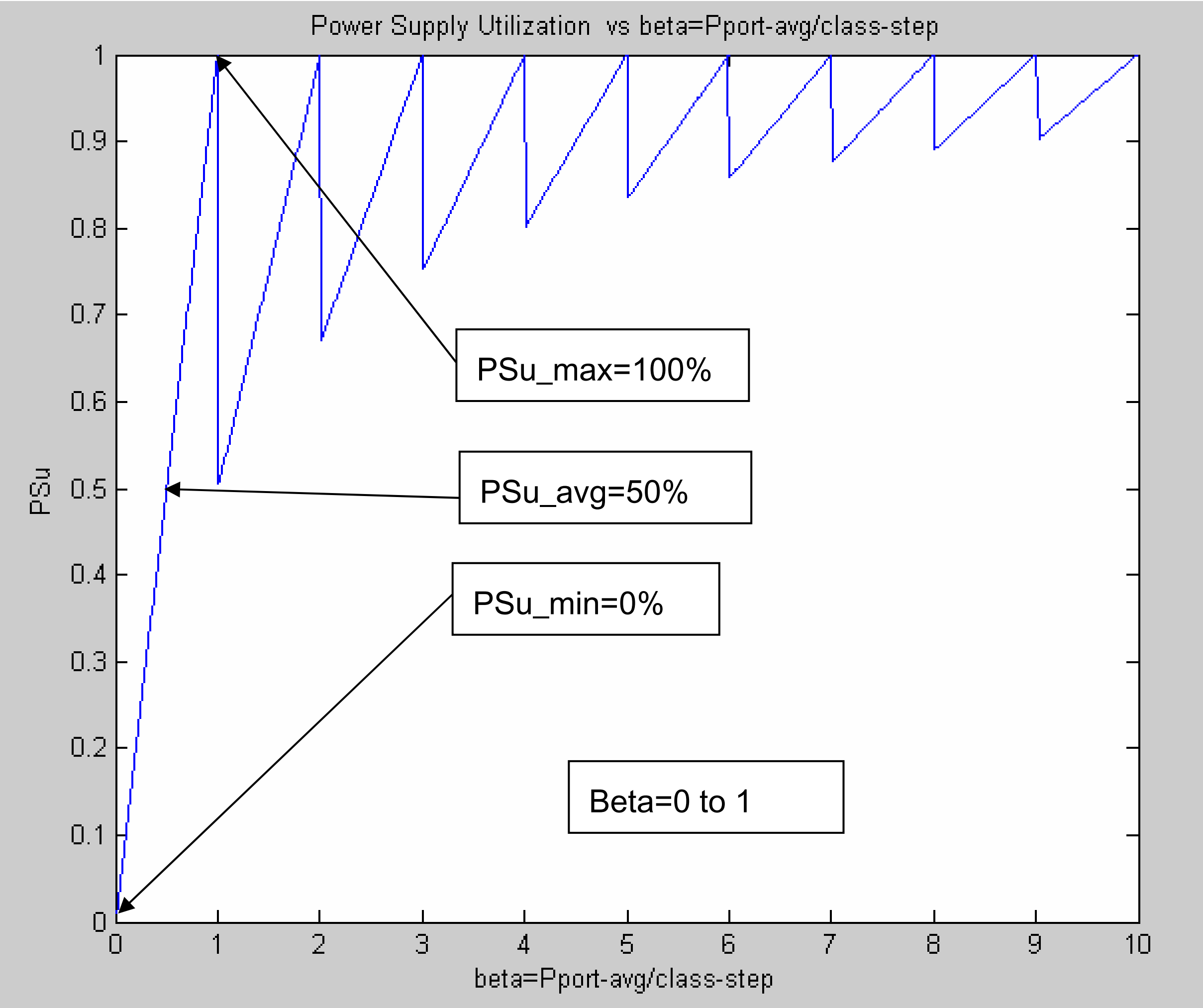

Figure 3: β=1 result with average PSu of 50%.

PSu is kind of periodic function that reaches to 100% whenever β is integer and starts with higher minimum value every time β is increased..

6.2 Sensitivity analysis

Line	β min	β max	Psu avg	Δ(Psu avg)/ Δ (β)
1	0	1	0.500	
2	1	2	0.753	25.25%
3	2	3	0.835	8.25%
4	3	4	0.876	4.12%
5	4	5	0.901	2.47%
6	5	6	0.917	1.65%
7	6	7	0.929	1.18%
8	7	8	0.938	0.88%
9	8	9	0.945	0.69%
10	9	10	0.950	0.55%
11	10	11	0.955	0.45%
12	11	12	0.959	0.37%
13	12	13	0.962	0.32%
14	13	14	0.965	0.27%
15	14	15	0.967	0.24%
16	15	16	0.969	0.21%

Table 2: Sensitivity analysis by solving Eq-17; PSu_avg vs β and PSu_avg marginal improvements vs β.

It can be seen from the above table ΔPsu avg / Δβ is decreased as β is increased.
It means that:

-Increasing β has economical limit.
The contribution of β for increasing PSu_avg is decreased as β increased beyond β_c.
β_c is the β in which the cost of increasing β equal to the cost savings of increase PSu.

- PSu average can never be 100% since β can never be infinite.

7. How to determine Pclass_avg or Pport_avg by the Marketing. Proposal

7.1 As we discussed previously, we don't know the probability function of Pport.
However we can make a list of current and future needs based on the experienced and limitations of the IEEE802.3af systems.

7.2 From a list of possible N port devices (1, 4, 6, 8, 12, ……N ports) we can access what is the power budget we willing to support in each device hence Pport_avg can be calculated for each case.

The worst case for us that the classification mechanism would need to support is the minimum Pport_avg obtained.

7.3 An example of the considerations that are needed to be taken in 7.2 are:
7.1 Pbudget is given (function of cost and system working environments)
7.2 The number of the ports in the system, N (function of usage, end customer, organization size etc.)
7.3 The power management statistics that we want to support; k1 and k2.
K1=the % of the ports that will be supported with maximum power.
K2=is the power for 1-K1 of the ports as percentage of the maximum power of the port.
K3= the probability that a PD will be connected to the port

From the above the system supports

Pport_avg= Pbudget / N =k3* (k1*N*Pport_max +(1-k1)*N*k2*Pport_max)/N =

<u>Pport_avg=Pport_max *k3*(k1*N* +(1-k1)*N*k2)/N</u>

Set β according to PSu required In order to support the Pport_avg required,
select Class_step<=Pport_avg/(PSu_avg*ceil(β))

7.2 Repeat the process of 7.1 to 7.3 for a list of current and future combinations of Pport_max, k1, k2 and N.

Note: there are other ways to asses minimum Pport_avg that we wish to support. It will be discussed in Part B of this paper.

<u>Examples:</u>

<u>E-1</u>
If we want to support N=4 port a system with:
Pport_max=16W.
All ports get max power therefore k1=1.
K2=0
K3=1
Pbudget=16W(1*4+0)=64W. If this number acceptable then power management is not required therefore we don't care about the classification parameters.

<u>E-2</u>
If we want to support N=48 port a system with:
Pport_max=40W.
K1=0.1. (10% of the ports get max power)
K2=0.2. (90% of the ports get 0.2*40W=8W)
K3=0.5 (Probability of 50% that a PD will be connected to the port)
Than:
Pbudget=0.5*40*(0.1*48+0.9*48*0.2)=269W
Pport_avg=269/48=5.6W.
Assuming we need PSu_avg=90% ==> β=5
Class_step=5.6/(0.9*5)=1.24W
Pclass_avg=ceil(5.6/1.24)*1.24=5.6W which is OK (guaranties that PSu can get 100% but the PSu average is 90%.
Number of classes=40/1.24=32-33.

7.2 Recommended design procedure based on the above analysis.

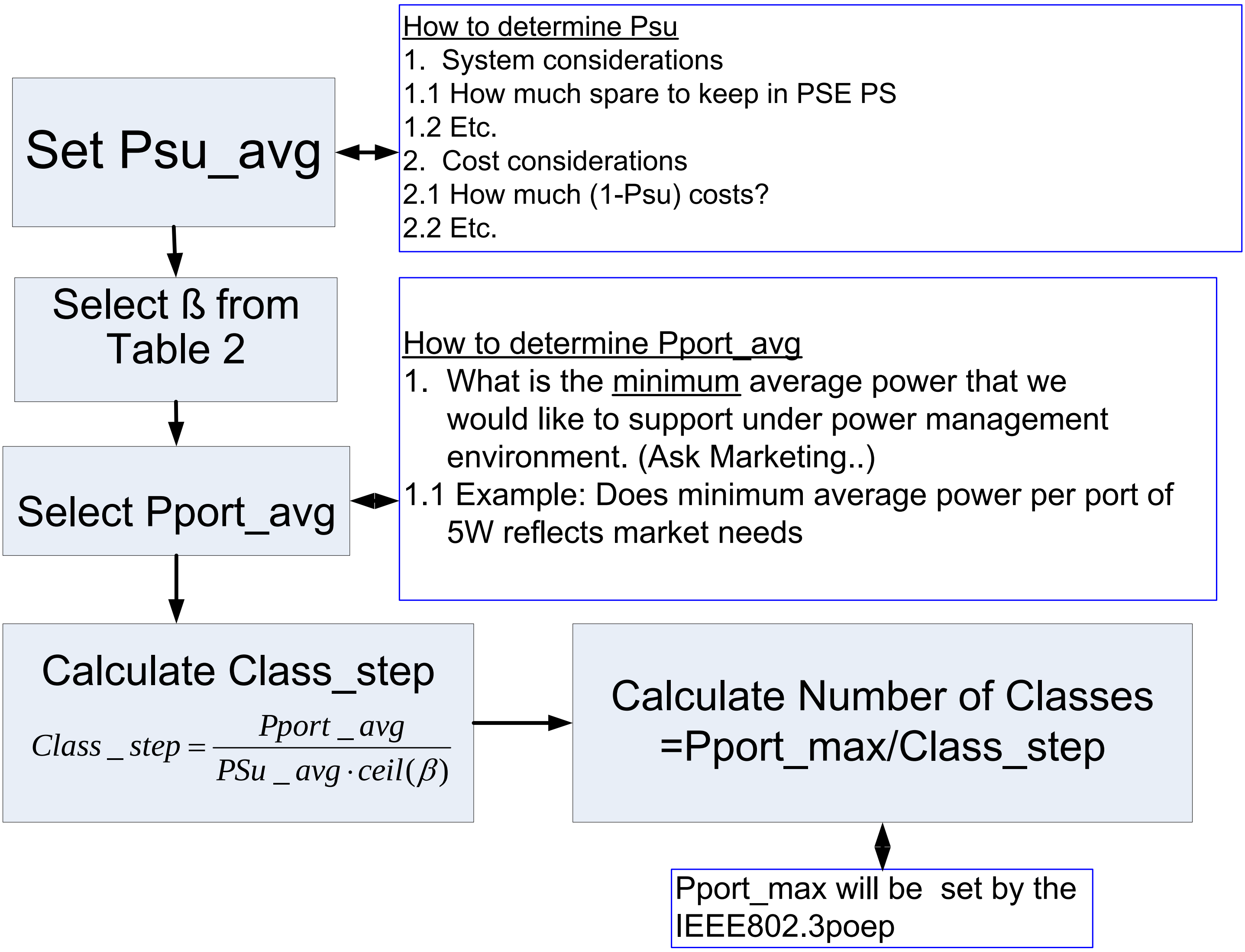

Figure - 4: Proposed Design Procedure and mapping of the unknowns

8. Determine number of classes

Number of classes is simply: Pport_max/Class_step for the linear step case.

If we want to support Pport_max=50Watt and if class_step is 1.6W then number of classes is: 31 – 32 classes.

9. Implementation issues

Example: If we wish to address 50W power range with 1mW power step, it requires 50,000 classes which will be excellent from power management point of view however will totally not be cost effective to implement in a chip from the following reasons:

1. Higher resolution costs with accurate references, comparators, A/D etc.
2. Currently, chip vendors satisfied with circuitry accuracy of 10%-20% which considered as cost effective.

2.1. 5-10% margins between classes was also chosen at the IEEE802.3af as design input and actually implemented in the standard.

The above considerations require that in any classification domain whether it is current (as in IEEE802.3af) or time, we will need design margin in each half side of the valid class region resulting with doubling the class region.
(Assuming that classification data is not passed from PD to PSE as hard constant).

The practical consequence is that the effective classes' region will increase from Pport_max/Class_step to 2Pport_max/Class_step.

If the classification domain is current (as we had used in the IEEE802.3af) and we would like to have 40 classes (just an example), with 5mA valid class region, then we would need 5mAx40x2=400mA current working region which is not practical due to thermal limitation at the PD chip. In this case coding techniques will reduce current working region.

If time domain is used (measuring current level with time duration) and we would like to have 40 classes, each with minimum duration of 5ms than we would need including margin for accuracy issues : 40x5msx2=400ms. Again, to reduce the time operating region we can code it by using different current levels which will reduce the time region required.

More work need to be done regarding clause 9, the implementation issues, however the above analysis is needed prior to looking for the implementation concept.
It is needed to set the specifications for the future classification method and can later help us to filter classification concepts that are not addressing our objectives.