



GRAU D'ENGINYERIA TELEMÀTICA

Assignació de recursos en xarxes
d'àrea corporal sense fils SmartBAN

PERE ARBÓS PARETS

Tutor

Jaume Ramis Bibiloni

Escola Politècnica Superior
Universitat de les Illes Balears
Palma, 11 de setembre de 2019

SUMARI

Sumari	i
Acrònims	iii
Resum	v
1 Introducció	1
2 L'estàndard ETSI SmartBAN	3
2.1 Construcció dels paquets de capa física	5
3 Modelat i algoritme de planificació per a xarxes SmartBAN	7
3.1 Modelat de la xarxa	7
3.2 L'eeAlgorithm	8
4 Teoria de Scheduling	11
4.1 Preliminars de la teoria de Scheduling	11
4.2 Model de <i>Scheduling</i> per a la minimització del Període d'accés planificat (SAP)	13
4.3 Resolució del model	16
4.3.1 Proposta de distribució de tasques en el model	16
4.3.2 Exemple de resolució	19
5 Anàlisi de resultats	23
5.1 Rendiment energètic i temps de bateria dels sensors	23
5.2 Simulació de la xarxa	25
5.3 Modes de transmissió	26
5.3.1 Variació en el nombre de repeticions	27
5.3.2 Millora en l'ús de minimització de T_{SAP}	27
5.3.3 Ús de BCH	29
5.4 Algoritme de selecció de mode de transmissió	29
6 Conclusions	33
Bibliografia	35

ACRÒNIMS

WBAN Wireless Body Area Network

IBI Inter Beacon Interval

SAP Període d'accés planificat

CMAP Període de control i gestió

LFJ Tasca Menys Flexible

IFS Inter Frame Space

MAC Medium Access Control

PDU Protocol Data Unit

PPDU PDU de capa física

MPDU PDU de capa MAC

SDU Service Data Unit

PSDU SDU de capa física

TM Mode de Transmissió

RESUM

Ens trobam en una situació en la que l'abastiment de recursos en l'àmbit sanitari és limitat, mentre que de cada cop hi ha més pacients crònics potencials, donat l'augment del percentatge de persones majors dins la població total. Per solucionar aquest problema s'ha proposat l'ús de xarxes d'àrea corporal sense fils (WBAN) de sensors que monitoritzen l'estat d'aquests pacients. És important cuidar el consum d'energia dels sensors que componen aquestes xarxes perquè puguin funcionar el major temps possible sense alimentació elèctrica externa. En el present treball s'estudien els mètodes perquè aquests sensors transmetin les seves dades de manera eficient.

Les xarxes SmartBAN són les xarxes WBAN que es defineixen a l'estàndard ETSI. En aquest estàndard, els sensors organitzen les seves transmissions en distints intervals anomenats Inter Beacon Interval (IBI). Dins cada IBI, s'organitzen les transmissions dels sensors de la xarxa en distints slots de durada fixa. Cada sensor sap a quins slots ha de transmetre abans de cada IBI, de manera que no hi ha col·lisions.

Per organitzar les transmissions a les xarxes SmartBAN es proposa l'eeAlgorithm. L'ús d'aquest algoritme resulta en un consum eficient d'energia per els sensors, que s'aconsegueix allargant el màxim possible la durada dels intervals IBI. Aquests resultats, però, són millorables optimitzant la distribució dels slots de transmissió. Mitjançant tècniques de teoria de *Scheduling*, es pot obtenir una distribució de slots òptima repartint els slots entre els distints intervals IBI de la manera més equitativa possible. D'aquesta manera, es pot allargar la durada de l'IBI mentre se segueixen complint els requeriments per el retard màxim de les dades a enviar.

INTRODUCCIÓ

En els darrers anys ens estam trobant amb un problema de manca de recursos i personal en els hospitals i centres mèdics per a proporcionar una deguda atenció sanitària a les persones que la necessiten. Com s'explica a [1] i [2], una de les raons d'aquesta situació és l'increment de població de les persones majors. El percentatge que suposen les persones majors sobre la població total és de cada cop més gran, i es tracta d'un grup de persones que solen requerir atenció mèdica de manera més freqüent que la resta. Amb la intenció d'aprofitar els avanços tecnològics per ajudar a solucionar aquesta problemàtica, s'han proposat diversos sistemes per mesurar la condició de les persones necessitades mitjançant l'ús d'aparells electrònics, i així reduir l'atenció sanitària personal que requereixen.

Entre aquests sistemes, l'ús de xarxes d'àrea corporal sense fils (WBAN, de l'anglès Wireless Body Area Network) és una de les propostes que ha despertat més interès entre els investigadors. Aquestes xarxes consisteixen en una sèrie de sensors que monitoritzen diferents paràmetres sobre l'estat del pacient i comuniquen les dades recopilades a un hub central que controla la xarxa. D'aquesta manera el hub pot comunicar l'estat del pacient al servidor del seu centre mèdic, per exemple per Internet. Això facilita en gran mesura la feina del personal del centre mèdic. Tot i així, l'aplicació de les xarxes WBAN no es limita a l'àmbit mèdic. Poden resultar també d'utilitat per al seguiment de l'entrenament d'un esportista, per exemple.

La comoditat de l'usuari és un aspecte important a considerar per a les xarxes WBAN. Els sensors han de ser petits i anar acoblats còmodament a la roba o al cos de l'usuari. També han de poder funcionar un temps suficientment llarg sense rebre alimentació elèctrica externa, ja que la condició de que funcionin sense fils és indiscutible. Aquestes característiques especials dels sensors, fan que es requereixi d'un estàndard per a xarxes WBAN que cuidi sobretot el consum d'energia dels sensors. L'estàndard que millor s'adapta a aquestes necessitats és l'estàndard proposat per l'ETSI ([3] i [4]).

Les xarxes WBAN que funcionen seguint l'estàndard ETSI s'anomenen xarxes SmartBAN. L'objectiu d'aquest treball és l'estudi de les xarxes SmartBAN i dels algorismes proposats per a la seva implementació.

L'eeAlgorithm proposat a [5] suposa una millora considerable en el consum d'energia dels sensors respecte de les propostes d'altres autors. Tot i així, aquests resultats segueixen sent millorables organitzant les transmissions dels sensors de manera òptima.

Aquest treball es basa principalment en la feina desenvolupada a [5]. L'aportació principal de l'alumne consisteix en la millora proposada per a l'eeAlgorithm. Aquesta millora s'aconseguirà modelant el problema de la distribució dels slots de transmissió dels sensors com un problema de *Scheduling*. A partir d'aquest model, es demostrarà que és possible trobar una distribució de slots òptima, i que aquesta distribució efectivament millora el consum d'energia dels sensors respecte de la versió original de l'eeAlgorithm.

L'estructura de la present memòria és la següent. Al capítol 2 s'explica el funcionament de l'estàndard ETSI. Després, s'introdueix un model matemàtic d'una xarxa SmartBAN basat en aquest funcionament al capítol 3, on també s'explica l'eeAlgorithm. Al capítol 4 s'introdueix la teoria de *Scheduling* i s'estudia la possibilitat de millora de l'eeAlgoritihm mitjançant el seu ús. Finalment, al capítol 5 s'analitzen els resultats obtinguts de l'aplicació de l'algoritme resultant en distints casos de xarxes SmartBAN.

L'ESTÀNDARD ETSI SMARTBAN

A l'estàndard ETSI per a xarxes SmartBAN hi trobam definits dos tipus de dispositius. Tenim per una banda els nodes, que són els sensors mèdics que generen informació de manera periòdica (o també fora d'aquest període en cas d'emergència); i per l'altra, un hub central que s'encarrega de rebre aquesta informació i d'organitzar la xarxa. Una xarxa SmartBAN està formada per un hub central i un conjunt de nodes (almenys un) que segueixen una topologia d'estrella (figura 2.1). Les connexions entre un hub i un node es realitzen sempre sobre un medi sense fils.

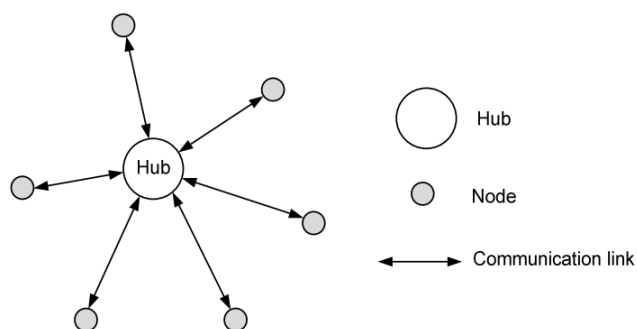


Figura 2.1: Topologia estrellada de les xarxes SmartBAN

La comunicació entre el hub i els nodes té lloc a dos canals diferents: un canal de control i un canal de dades. Al canal de control, el hub central s'encarrega de transmetre periòdicament una trama anomenada C-Beacon. Aquesta trama conté informació sobre diversos paràmetres importants per a la comunicació en el canal de dades. No hi ha més transmissions en el canal de control (els nodes no hi transmeten).

Al canal de dades hi transmeten tant el hub com els nodes. Aquest canal es troba dividit en diversos intervals anomenats IBIs, de duració T_{IBI} . Cada un d'aquests intervals s'organitza al seu cop en diversos períodes de transmissió, que són els següents (figura 2.2):

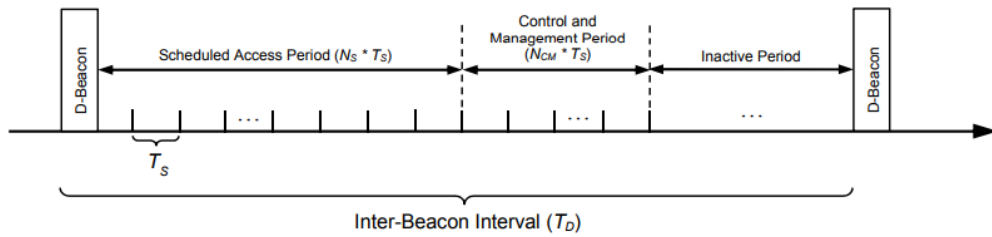


Figura 2.2: Esquema d'un interval IBI

- *D-Beacon*: El hub central transmet una trama anomenada D-Beacon que senyalitza l'inici de cada IBI.
- *Període d'accés planificat (SAP)*: En aquest període els nodes transmeten les dades que generen de manera periòdica. Està dividit en N_S slots de duració T_S , i cada un d'aquests slots s'empra per a la transmissió d'un paquet d'informació d'un node en concret de la xarxa. Cada node tindrà assignat un cert nombre de slots dins aquest període segons el volum i la periodicitat de les dades que genera. Havent planificat prèviament la distribució d'aquests slots, cada node sap a quin instant li toca transmetre quants de paquets, i per tant no es necessita cap altre protocol d'accés al medi. La transmissió dins aquests slots es realitza segons la figura 2.3.

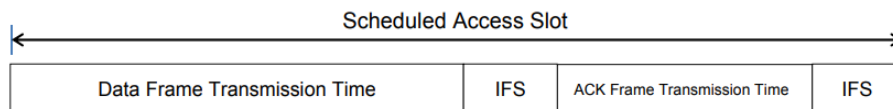


Figura 2.3: Composició dels slots del període d'accés planificat

La duració T_S dels slots ha de ser tal que $T_S = l \cdot T_{min}$, amb $T_{min} = 625 \mu s$ i $l = 2^b$, $b \in \{0, 1, 2, 3, 4, 5\}$. El període d'accés planificat té una duració $T_{SAP} = N_S \cdot T_S$

- *Període de control i gestió (CMAP)*: És on els dispositius transmeten diferents tipus de trames relacionades amb la gestió de la xarxa. En aquest període, els nodes que vulguin sol·licitar slots de transmissió dins el període d'accés planificat envien una trama C-Req al hub central. El hub contesta amb una trama C-Ass, que indica els slots que s'han assignat al node. Si en qualche moment es requereix una reassignació d'aquests slots, el hub ho comunicarà a la següent trama D-Beacon. Aquest període CMAP també s'empra per transmetre paquets d'emergència que contenen informació urgent generada pels sensors de manera imprevisible. L'accés al medi en aquest període es pot realitzar seguint un protocol d'ALOHA ranurat o bé Multi-Use Channel Access. Té una duració T_{CMAP} .
- *Període inactiu*: Període on cap dispositiu hi transmet i de duració T_{INP} ajustable segons convengui.

2.1 Construcció dels paquets de capa física

A continuació s'explica el procés a seguir per obtenir els paquets que s'envien per capa física a partir de les trames de nivell MAC. Partint d'una trama MAC de longitud L_f , primer de tot se li afegeix una capçalera de 48 bits, juntament amb 16 bits de paritat, per obtenir la Protocol Data Unit (PDU) de capa MAC, la MPDU. En cas de que sigui necessari, l'estàndard ofereix l'opció d'afegir a aquesta MPDU una codificació BCH(127, 113). Un cop es decideix si es codifica o no la MPDU, el resultat forma la Service Data Unit (SDU) de capa física (PSDU). Finalment, a la PSDU s'hi afegeixen una capçalera de capa física (PLCP) de 40 bits i un preàmbul de 16 bits per obtenir el paquet final, que serà la PDU de capa física (PPDU). La figura 2.4 il·lustra aquesta estructura.

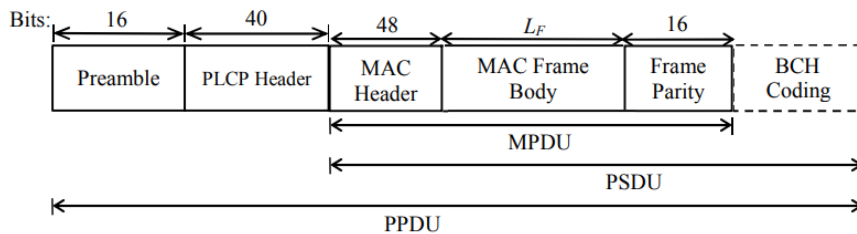


Figura 2.4: Estructura de la PDU de capa física

Per millorar la probabilitat d'error a l'hora de transmetre un paquet, l'estàndard ofereix l'opció de repetir la transmissió de la PDU de capa física. D'aquesta manera, la PDU de capa física se podrà transmetre un nombre N_{REP} de vegades, amb $N_{REP} \in \{1, 2, 4\}$.

MODELAT I ALGORITME DE PLANIFICACIÓ PER A XARXES SMARTBAN

3.1 Modelat de la xarxa

Necessitam un model d'una xarxa SmartBAN amb paràmetres concrets. Partim d'una xarxa SmartBAN de N nodes amb les següents característiques:

- El conjunt de períodes $T_G = \{T_{g_1}, \dots, T_{g_N}\}$, on T_{g_i} indica el temps que tarda el node i en generar dades per transmetre.
- El conjunt de volums de dades $V = \{V_1, \dots, V_N\}$, on V_i indica el nombre de bits de dades que genera el sensor i cada T_{g_i} .
- El conjunt de retards $D = \{D_1, \dots, D_N\}$, on D_i indica el temps màxim en què es poden rebre les dades del sensor i un cop les genera.
- El retard màxim permès per a dades d'emergència, D_E (el mateix per a tots els nodes).

Consideram ara un node arbitrari de la xarxa, i . Volem trobar el temps de slot, T_{S_i} que necessita per transmetre al període d'accés planificat. Tenim que el volum de les dades que genera aquest node i és V_i . Seguint el que s'ha explicat a la secció anterior (2.1), afegint els bits de capçalera i de paritat, obtenim la longitud de la MPDU:

$$L_{MPDU_i} = V_i + L_{MACHdr} + L_{paritat} = V_i + 64 \text{ [bits]} \quad (3.1)$$

Per obtenir la longitud de la PSDU, cal considerar si s'utilitza o no codificació BCH:

$$L_{PSDU_i} = \begin{cases} L_{MPDU_i}, & \text{sense cod.} \\ L_{MPDU_i} + \left\lceil \frac{L_{MPDU_i}}{k} \right\rceil (n - k), & \text{amb cod. BCH}(n,k) \end{cases} \quad (3.2)$$

Així, la longitud de la PPDU s'obté afegint els bits del preàmbul i de la capçalera de capa física:

$$L_{PPDU_i} = L_{PSDU_i} + L_{preàmbul} + L_{PLCPHdr} = L_{PSDU_i} + 56 \text{ [bits]} \quad (3.3)$$

Per tant el temps de transmissió de la PPDU serà:

$$T_{DATA_i} = \frac{L_{PPDU_i}}{R} \cdot N_{REP_i} \quad (3.4)$$

On N_{REP_i} és el valor de N_{REP} per al sensor i i R la taxa de transmissió $R = 1\text{Msímbol/s}$ definida a l'estàndard. Com que hem considerat apart els bits de codificació BCH i les repeticions de PPDU, tenim que 1 símbol = 1 bit.

Basta afegir-hi el temps de transmissió d'ACK i els dos períodes IFS (figura 2.3, capítol 2) per obtenir el temps de slot per al sensor i . Seguint l'estàndard tenim que aquests valors són $T_{ACK} = 0.12 \text{ ms}$ i $T_{IFS} = 0.15 \text{ ms}$.

$$T_{S_i} = T_{DATA_i} + T_{ACK} + 2T_{IFS} = T_{DATA_i} + 0.42 \text{ [ms]} \quad (3.5)$$

Tenim que l'equació 3.5 es compleix per a tots els nodes. Si suposam que la xarxa admet fragmentació de les trames MAC, i consideram el mínim dels temps de slot per a tots els sensors, $T_{S_{min}} = \min \{T_{S_i} \mid i \in \{1, \dots, N\}\}$, només cal ajustar el temps de slot a un dels valors vàlids definits a l'estàndard. Per tant el temps de slot dins el període d'accés planificat serà

$$T_S = \min \{T \in \mathbb{R} \mid T \geq T_{S_{min}} \wedge T = lT_{min}, T_{min} = 625\mu\text{s}, l = 2^b, b \in \{0, 1, 2, 3, 4, 5\}\} \quad (3.6)$$

A partir d'aquí, també podem obtenir el nombre de slots en què es fragmentaran les dades d'un sensor arbitrari i com

$$l_{S_i} = \left\lceil \frac{T_{S_i}}{T_S} \right\rceil \quad (3.7)$$

3.2 L'eeAlgorithm

Es requereix d'un algoritme que organitzi les transmissions dels sensors dins el SAP i estableixi la durada dels intervals IBI. L'objectiu és, doncs, aconseguir que els intervals IBI durin el màxim de temps possible mentre que es compleixen les restriccions de la xarxa. Maximitzar la durada de l'IBI és important perquè, segons els estudis realitzats a [6], una major durada de l'IBI implica un major rendiment dels sensors i per tant un augment en el temps de bateria dels mateixos.

Partim de l'algoritme proposat a [5], l'anomenat eeAlgorithm. La idea principal d'aquest algoritme és agrupar tantes transmissions com es pugui d'aquells sensors que tenen un període de generació de dades més petit, sempre que es compleixin els requisits del retard. D'aquesta manera, s'estableix el conjunt de períodes de transmissió $T_t = \{T_{t_1}, \dots, T_{t_N}\}$, on T_{t_i} indica cada quant de temps li toca al node i transmetre dades. També podem obtenir el nombre σ_i de slots que requerirà el node i per transmetre les dades que té pendents com $\sigma_i = \frac{T_{t_i}}{T_{g_i}} \cdot l_{S_i}$ (com a resultat de l'algoritme, T_{t_i} sempre

serà múltiple del període de generació de dades T_{g_i}). El diagrama de flux original de l'eeAlgorithm es troba a la figura 3.1

A continuació s'explica la versió final de l'algoritme passa a passa. La modificació i l'aportació principal d'aquest treball es troba a la passa 3.

- *Passa 0:* Consideram tots els paràmetres exposats al modelat de la xarxa de la secció anterior. Prenem el conjunt de retards D com els temps d'espera definitius per a transmetre les dades que generen els sensors (no tenim en compte possibles retards causats per l'entorn de la xarxa o al servidor destí). Inicialitzam $T_{IBI} = T_{IBI}^{max} = \min(D)$.
- *Passa 1:* Per a cada node i , $i \in \{1, \dots, N\}$ obtenim el seu període de transmissió com $T_{t_i} = T_{g_i} P_i$. P_i indica el nombre de transmissions agrupades del node i , i s'obté com $P_i = \frac{T_{IBI}}{MCD(T_{g_i}, T_{IBI})}$.
Cal observar que T_{t_i} també és múltiple de T_{IBI} per a qualsevol node, perquè $T_{t_i} = \frac{T_{g_i}}{MCD(T_{g_i}, T_{IBI})} \cdot T_{IBI}$, on $\frac{T_{g_i}}{MCD(T_{g_i}, T_{IBI})}$ és clarament enter ja que $MCD(T_{g_i}, T_{IBI})$ divideix T_{g_i} .
- *Passa 2:* Verificam per a cada node que el seu període de transmissió no supera el retard màxim permès, i.e. $T_{t_i} \leq D_i \forall i$. Si es compleix, continuam. Si no, anam a la passa 6.
- *Passa 3:* Consideram la supertrama IBI de durada $mcm(T_t)$, formada per un total de $\frac{mcm(T_t)}{T_{IBI}}$ intervals IBI. L'objectiu ara és organitzar les transmissions pendents dels nodes dins aquesta supertrama de manera que es minimitzi la durada del SAP a cada IBI. Aquest problema es resol modelant-lo com un problema de *Scheduling* i s'explica en profunditat al capítol 4.
- *Passa 4:* Hem de verificar ara que el tràfic d'emergència es pot enviar a temps. Aquest tràfic el transmeten els nodes durant el CMAP, per tant si consideram el pitjor cas en què aquest tràfic es genera a l'inici del període inactiu, tenim que s'ha de complir $T_{SAP} + T_{InP} \leq D_E$. Com que la durada del període inactiu pot ser arbitràriament petita, comprovam que la durada del SAP obtinguda a la passa anterior compleix $T_{SAP} < D_E$. Si es compleix, continuam. Si no, anam a la passa 6.
- *Passa 5:* Establim les durades del CMAP i del període inactiu segons:
 - Si $T_{IBI} > D_E$, aleshores $T_{CMAP} = T_{IBI} - D_E$
 - Altrament, si $T_{IBI} > 2 \cdot T_{SAP}$, aleshores $T_{CMAP} = T_{SAP}$

$$i T_{InP} = T_{IBI} - T_{SAP} - T_{CMAP}.$$

L'algoritme acaba aquí.

- *Passa 6:* Si hem arribat a $T_{IBI} = T_{IBI}^{min} = MCD(T_G)$, vol dir que els requeriments no es poden complir en aquesta xarxa i acaba l'algoritme. Si no, actualitzam $T_{IBI} = T_{IBI} - 1$ i tornam a la passa 1.

3. MODELAT I ALGORITME DE PLANIFICACIÓ PER A XARXES SMARTBAN

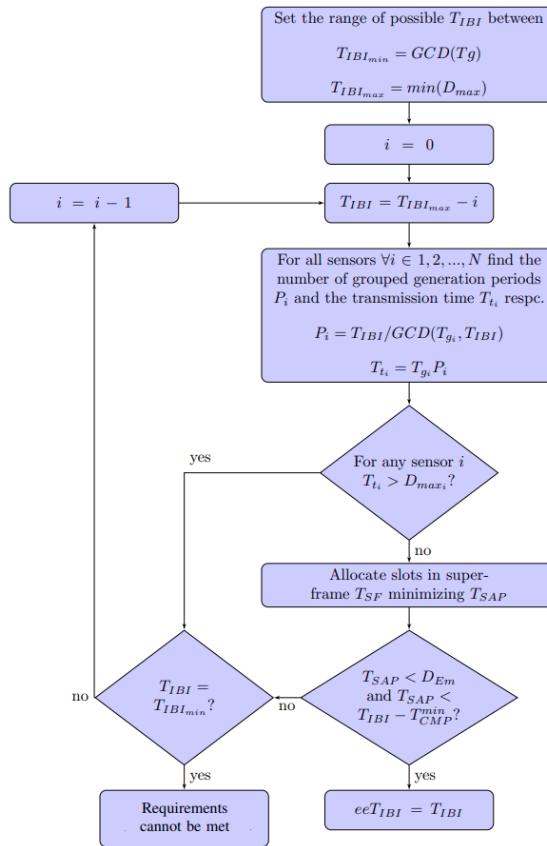


Figura 3.1: Diagrama de flux de l'eeAlgorithm

TEORIA DE SCHEDULING

Un cop s'han establert la durada dels intervals IBI i el conjunt de períodes de transmissió T_i dins l'eeAlgorithm, volem planificar les transmissions dels diferents nodes de manera que es minimitzi la durada del SAP dins aquests IBI. En aquest capítol es planteja aquesta problemàtica com un problema de *Scheduling* i es tractarà de resoldre mitjançant la teoria desenvolupada a [7].

4.1 Preliminars de la teoria de Scheduling

Explicam algunes de les nocions bàsiques sobre la teoria de *Scheduling* abans de presentar el model del nostre problema. Aquests conceptes es troben explicats en profunditat a [7], i és també la referència principal sobre el que s'exposa en aquesta secció.

Un problema de *Scheduling* consisteix en un conjunt de **tasques** a realitzar o completar juntament amb una sèrie de recursos que permeten realitzar aquestes tasques. L'objectiu és trobar una manera de distribuir aquests recursos entre les distintes tasques complint una sèrie de restriccions i optimitzant un cert paràmetre resultant; com per exemple el temps total que es triga en realitzar totes les tasques, o el retard de les tasques que no es completen en un temps límit. Aquests recursos que ens permeten realitzar les tasques es modelen com a **màquines**. D'aquesta manera, l'esmentada distribució de recursos consisteix en assignar a cada màquina una o diverses tasques (amb un ordre concret en cas de que sigui rellevant), fins que totes les tasques hagin estat assignades. Cada màquina completa les tasques que se li assignen una rere l'altra. En un model de *Scheduling*, hi pot haver una o diverses màquines, i aquestes poden ser iguals o tenir característiques diferents.

Primer de tot, ens trobam amb què els models de *Scheduling* es divideixen en dos tipus: els models deterministes i els models estocàstics. Els models deterministes tracten amb un nombre finit de tasques, les característiques de les quals són conegudes prèviament. Als models estocàstics, en canvi, les tasques a realitzar presenten

característiques que poden ser aleatòries, com el temps de processament necessari per realitzar-les o la seva data límit. En aquest cas, s'empren variables aleatòries per definir el comportament d'aquests paràmetres. En el nostre problema, els paràmetres requerits per al nostre model de *Scheduling* són coneguts, i per tant ens centram en els models deterministes.

En un model de *Scheduling*, les tasques a realitzar solen tenir característiques importants a tenir en compte. Aquestes característiques es tracten, principalment, dels següents paràmetres.

- El *temps de processament* que tarda una tasca en completar-se. Aquest temps pot dependre o no de la màquina a la que se li assigna la tasca.
- La *data de llançament* de la tasca. Indica el moment en què la tasca arriba al sistema, és a dir, l'instant de temps a partir del qual es pot començar el seu processament.
- La *data límit* per a completar la tasca. Les dates límits poden ser definitives en el sentit de que el problema té com a restricció que totes s'han de complir. Però també potser que simplement impliquin una certa penalització en cas d'incomplir-se, i que l'objectiu del problema sigui minimitzar aquestes penalitzacions.
- El *pes* o la importància que té una tasca respecte d'altres. Aquest paràmetre es pot traduir en diversos factors, com per exemple la magnitud de l'esmentada penalització en cas d'incomplir la data límit de la tasca.

Un problema concret de *Scheduling* es pot descriure mitjançant una tripleta com $\alpha | \beta | \gamma$. El camp α indica el sistema de màquines del problema: quantes n'hi ha i si tenen característiques especials. β pot indicar alguns paràmetres de les tasques a realitzar, o altres restriccions del problema. Aquest camp pot incloure cap, una o diverses entrades. El camp γ indica el paràmetre o funció a minimitzar. Aquesta funció sempre sol ser una acumulació dels retards de les tasques o el temps total en completar-les totes, per tant el tipus d'optimització desitjada és la minimització.

Alguns exemples de sistemes de màquines que es poden expressar al camp α són:

- *Màquina única*. Totes les tasques es realitzen a la mateixa màquina. En aquest cas l'objectiu es trobar l'ordre a seguir per completar les tasques. S'indica a la tripleta com a 1.
- *Màquines idèntiques en paral·lel*. Tenim un nombre m de màquines idèntiques. Cada tasca s'ha de completar en una, i només una, de les màquines. Es pot donar la restricció de que cada tasca només es pugui completar en un cert subconjunt de les màquines. S'indica com a Pm .
- *Màquines no idèntiques en paral·lel*. Es tracta d'una generalització del cas anterior on cada màquina té una velocitat de processament diferent. Per exemple, si la tasca j té un temps de processament p_j , i la màquina i una velocitat v_i , aleshores el temps que tardarà la màquina i en completar la tasca j serà p_j/v_i . Aquest cas també es pot generalitzar al seu cop considerant que les velocitats de les màquines depenen de la tasca que estan processant. S'indica com a Qm en cas

de que les velocitats no depenen de les tasques, i com a Rm en cas de que sí (on m torna a ser el nombre de màquines).

Existeixen també altres sistemes de màquines més complicats, on per exemple les tasques s'han de completar en més d'una màquina, el que fa que cada tasca segueixi una ruta de diferents màquines. El grau de flexibilitat d'aquestes rutes pot variar.

Al camp β s'indica si intervenen en el problema alguns paràmetres de les tasques a realitzar, com el temps de processament o la data límit. Per exemple, si volem indicar que els temps de processament de totes les tasques són iguals, inclouríem en aquest camp l'entrada $p_j = 1$. En aquest camp també s'inclouen possibles restriccions a tenir en compte per resoldre el problema. Aquestes restriccions poden ser molt variades, però alguns exemples de les més comuns serien:

- *Precedències*. Es pot donar el cas que per començar a processar una tasca s'hagin d'haver completat abans una o diverses altres tasques. Un exemple típic d'aquesta restricció seria la realització d'un projecte. S'indica amb l'entrada *prec* al camp β .
- *Retards en funció de la seqüència*. Estam parlant del cas en què una màquina ha d'esperar un cert temps per començar a processar la tasca k just després d'haver completat la tasca j . Això introdueix al sistema un retard s_{jk} , que pot dependre o no de la màquina que processa les tasques. S'indica amb l'entrada s_{jk} .
- *Restriccions d'assignació de les màquines*. Aquesta restricció es pot donar en sistemes de màquines idèntiques en paral·lel. Es dóna quan no totes les màquines són capaces de processar totes les tasques. Per cada tasca j , tenim un conjunt M_j que inclou totes les màquines que poden processar la tasca j . S'indica amb l'entrada M_j .

Consideram per a cada tasca j l'instant de temps C_j en què aquesta es completa. El paràmetre a minimitzar especificat al camp γ , anomenat funció objectiu, sempre serà una funció d'aquests temps de finalització. Una funció objectiu típica seria el temps de finalització del sistema, C_{max} , que s'obté com el temps de finalització de la darrera tasca en completar-se, i.e. el màxim valor d'entre tots els C_j . Aquest cas s'indica amb l'entrada C_{max} al camp γ .

En cas de que les tasques tinguin data límit, hi ha diverses altres funcions objectiu a considerar. Per exemple, el nombre de tasques que sobrepassen la data límit, o bé la suma de retards de les tasques respecte la seva data límit. En aquest tipus de funcions hi solen aparèixer els pesos de les distintes tasques, de manera que sigui més important acabar a temps unes certes tasques respecte d'altres.

4.2 Model de *Scheduling* per a la minimització del SAP

Cercam ara una manera de modelar el problema de minimització de la durada del SAP dins l'eeAlgorithm com un problema de *Scheduling*. Recordam que teníem el conjunt T_t de períodes de transmissió de cada un dels N nodes de la xarxa, juntament amb la durada de l'interval IBI, T_{IBI} . A partir d'aquí es pot definir una supertrama IBI de

durada $mcm(T_i)$, que estarà formada per tant d'un total de $\frac{mcm(T_i)}{T_{IBI}}$ intervals IBI seguits (ja hem vist que els elements de T_i són múltiples de T_{IBI}).

Primer de tot cal identificar quins elements poden representar les tasques i les màquines del model de *Scheduling*. És bastant intuïtiu identificar de qualche manera les dades pendents que tenen els nodes per transmetre amb les tasques a realitzar. Hem vist que dins el SAP les transmissions dels diferents nodes s'organitzen en slots de duració T_S (paràmetre que ja s'ha calculat). Donat un node i , i a partir del seu període de generació de dades T_{g_i} i el nombre l_{S_i} de slots que ocupen les dades generades dins el SAP, obteníem el nombre σ_i de slots que requereix el node i cada T_{t_i} com $\sigma_i = \frac{T_{t_i}}{T_{g_i}} \cdot l_{S_i}$. Si definim els slots que s'han de transmetre dins la supertrama com les tasques a realitzar, tenim en el nostre model un total de $\sum_{i=1, \dots, N} \sigma_i$ tasques. Com que tots els slots tenen la mateixa durada, ens trobam en el cas especial en què el temps de processament de totes les tasques és el mateix.

Falta trobar els elements que actuaran com a màquines en aquest model. Si els slots de dades són les tasques a realitzar, els recursos que ens permeten completar aquestes tasques serien els diferents intervals IBI que formen la supertrama. Al cap i a la fi, l'objectiu era repartir les transmissions dels nodes entre els diferents IBI de la supertrama. Traduint aquests elements a tasques i màquines obtenim de manera prou natural un problema de *Scheduling*. El fet de que un node transmeti dades en un cert IBI, no afecta a la velocitat de transmissió, el que implica que totes les màquines del sistema són equivalents. Tenim per tant un sistema format per $m = \frac{mcm(T_i)}{T_{IBI}}$ màquines idèntiques en paral·lel.

Cal assegurar que les transmissions dels nodes es realitzin de manera que es compleixin les necessitats de la xarxa. Recuperant el model de la xarxa, teníem en el conjunt D del temps màxim que pot esperar cada node en enviar dades des de que les genera. A la segona passa de l'Algorithm, ja s'ha comprovat que $T_{t_i} \leq D_i$. Per tant basta assegurar que qualsevol node i pot transmetre els seus σ_i slots de dades en un temps inferior o igual a T_{t_i} .

Per aplicar aquesta restricció al problema, la solució intuïtiva podria ser l'ús de dates límits, o també l'ús de dates de llançament per assegurar que les dades no es poden transmetre abans de ser generades. El problema és que l'ús de dates de llançament pot donar lloc a un problema més complicat del necessari, mentre que l'ús de dates límits no acaba de garantir que aquestes es compleixin i no és del tot compatible amb la funció objectiu que voldrem minimitzar (com es veu més endavant). L'opció restant és aprofitar el fet de que les distintes màquines del sistema (els intervals IBI de la supertrama) estan totes separades en el temps. D'aquesta manera, la restricció aplicada al model consistirà en restringir les màquines que poden processar cada tasca. Les tasques corresponents als slots de dades que genera un node s'assignaran només a les màquines corresponents als intervals IBI que tenen lloc en el corresponent període de transmissió. Això resulta, per a cada tasca j , en el conjunt M_j de màquines que la poden processar, com s'ha vist a la secció anterior.

Per exemple, considerem un node i amb un període de transmissió $T_{t_i} = 300ms$ i que transmet a cada període $\sigma_i = 5$ slots de dades. La durada de l'IBI és $T_{IBI} = 100ms$ i la durada de la supertrama IBI és de $900ms$, de manera que està formada per nou IBIs, el que es tradueix a un model de nou màquines en paral·lel. Com que la supertrama ocupa tres períodes de transmissió del node i , aquest requerirà d'un total de 15 slots

per transmetre, el que defineix 15 tasques distintes (veim que si consideram les tasques definides per una xarxa de, diguem, sis o set nodes, el nombre total de tasques pot ser elevat). Com hem dit, hem d'assegurar que cada una d'aquestes tasques es transmet en el seu període de transmissió. Cada període de transmissió conté 3 IBI, i per tant cada tasca només es pot assignar a les tres màquines corresponents:

- Les primeres cinc tasques només es poden assignar a les tres primeres màquines. Aquesta restricció l'escriuríem com $M_1 = M_2 = M_3 = M_4 = M_5 = \{1, 2, 3\}$.
- Les següents cinc tasques només es poden assignar a les següents tres màquines: $M_6 = \dots = M_{10} = \{4, 5, 6\}$.
- I les darreres cinc tasques les restringiríem segons $M_{11} = \dots = M_{15} = \{7, 8, 9\}$.

Finalment, resta considerar la funció objectiu a minimitzar. L'objectiu del problema era minimitzar la durada del SAP. Això és equivalent a minimitzar el nombre de slots de transmissió que conté. Per comoditat consideram que tots els IBI contenen el mateix nombre de slots (si un IBI no se li assignen prou transmissions per omplir tots els slots, en aquests slots restants no hi ha transmissió). D'aquesta manera, el nombre de slots de cada IBI serà igual al màxim nombre de tasques que s'han assignat a una màquina. Aquest nombre serà la funció a minimitzar.

Com que totes les tasques tenen el mateix temps de processament, la funció objectiu equival al temps de finalització del sistema, C_{max} . Donat que les "màquines" del nostre model actuen en temps diferents, potser no sigui del tot clar considerar el temps de finalització com a la funció objectiu. Però si un pensa en els slots de transmissió com a les unitats de temps del sistema, i considera que les màquines actuen de manera paral·lela, en aquest cas, en el temps, es pot veure més clarament que C_{max} és la funció objectiu que ens interessa (4.1).

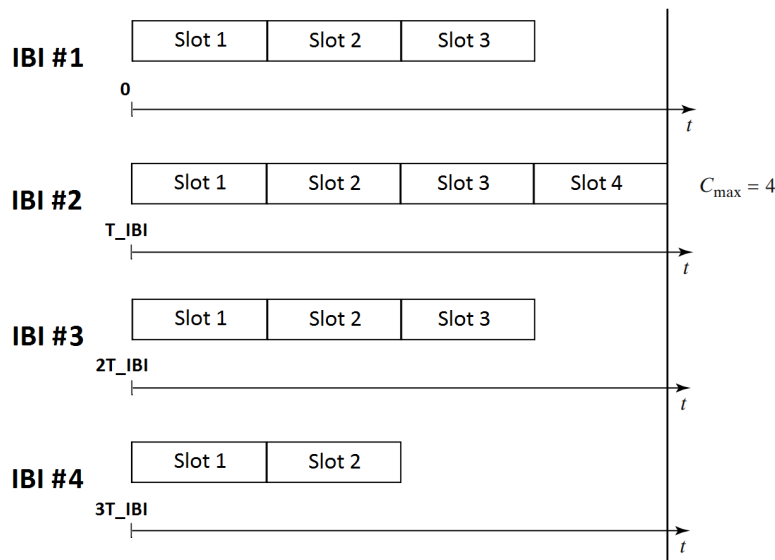


Figura 4.1: Exemple visual de modelat del problema

Concloem que el nostre problema de *Scheduling* és del tipus $P_m \mid p_j = 1, M_j \mid C_{max}$.

4.3 Resolució del model

El model de *Scheduling* obtingut és un dels que es tracta la seva resolució a [7]. En el llibre es demostra que l'anomenada regla de la Tasca Menys Flexible (LFJ, de l'anglès *Least Flexible Job*) obté una solució òptima per al problema $P_m \mid p_j = 1, M_j \mid C_{max}$, sempre que els conjunts M_j formin una col·lecció de conjunts encaixats (en anglès, *nested*). Es diu que els conjunts d'una col·lecció són encaixats quan per a qualsevol parella (A, B) de conjunts de la col·lecció es compleix que $A \subseteq B$, o bé $B \subseteq A$, o bé $A \cap B = \emptyset$.

La regla LFJ es tracta d'un algoritme senzill que assigna primer les tasques que tenen més restringit el conjunt de màquines que les poden processar. Cada instant de temps es tria una màquina desocupada qualsevol i , de les tasques que pot processar, se li assigna la que tenguí menys màquines al conjunt M_j . Si diverses tasques empaten, es tria una d'elles arbitràriament. Veim que és un algoritme bastant bàsic i que pot donar peu a millores.

Malauradament, els conjunts M_j del nostre model no són necessàriament encaixats. Vegem-ne un contraexemple. Considerem una xarxa amb $T_{IBI} = 100ms$ i una supertrama de 10 IBIs. Tenim dos nodes i i j amb períodes de transmissió $T_{t_i} = 200ms$ i $T_{t_j} = 500ms$. Ambdós nodes transmeten un únic slot de dades a cada període de transmissió. Això defineix un model amb 10 màquines, 5 tasques per part del node i , diguem-ne i_1, \dots, i_5 , i dues tasques per part del node j , j_1 i j_2 . Considerant les màquines que poden processar aquestes tasques, tenim que $M_{i_3} = \{5, 6\}$ i $M_{j_1} = \{1, 2, 3, 4, 5\}$. Aquests conjunts no compleixen la condició de conjunts encaixats. Per tant no podem garantir que la regla LFJ trobi la solució òptima per al nostre model.

Cal trobar una manera alternativa per a obtenir una distribució òptima de les tasques. En lloc de tractar d'obtenir un algoritme que resolgui el problema general $P_m \mid p_j = 1, M_j \mid C_{max}$ sense requeriments sobre la família de conjunts M_j , optem per cercar una solució senzilla aprofitant les peculiaritats d'aquests conjunts M_j resultants del nostre model.

4.3.1 Proposta de distribució de tasques en el model

Considerem primer una estructura general del model. Tenim una xarxa de N nodes amb intervals IBI de durada T_{IBI} .

- El node 1 transmet σ_1 slots de dades en un període de transmissió T_{t_1} .
- \vdots
- El node N transmet σ_N slots de dades en un període de transmissió T_{t_N} .

Com hem vist, això ens defineix una estructura de $m = \frac{mcm(T_i)}{T_{IBI}}$ màquines. Cada node i veu $\pi_i = \frac{mcm(T_i)}{T_{t_i}}$ períodes de transmissió dins la supertrama IBI, i per tant defineix $\pi_i \sigma_i$ tasques. Tenim aleshores un total de $\tau = \sum_{i=1, \dots, N} \pi_i \sigma_i$ tasques. Una possible observació seria trobar una fita per a la funció objectiu C_{max} . Si poguéssim assignar

les tasques a les màquines amb total llibertat, obtindríem un temps de finalització òptim assignant a cada màquina el mateix nombre de tasques. Si repartim les tasques equitativament, cap a $\frac{\tau}{m}$ tasques per màquina. Si aquest nombre no és enter, assignam $\lceil \frac{\tau}{m} \rceil$ tasques a una part de les màquines i $\lfloor \frac{\tau}{m} \rfloor$ a l'altra. Per tant tenim que la funció objectiu del nostre model compleix $C_{max} \geq \lceil \frac{\tau}{m} \rceil$.

Si ara trobam una distribució de tasques en el nostre model que resulti en $C_{max} = \lceil \frac{\tau}{m} \rceil$, haurem trobat una solució òptima per al problema. A continuació es proposa una distribució que ho aconseguim.

La idea principal és anar assignant juntament totes les tasques resultants de cada node. D'aquesta manera l'assignació de tasques es realitza en N passes. Sigui $\mu_i = \frac{T_i}{T_{IBI}}$ el nombre d'interval IBI que veu el node i dins cada un dels seus períodes de transmissió. Notis que $\pi_i \mu_i = m$. Aleshores, per a un node i qualsevol tenim:

- σ_i tasques que només es poden assignar a les primeres μ_i màquines, per tant els seus conjunts M_j són tots iguals a $\{1, \dots, \mu_i\}$.
- σ_i tasques amb conjunts M_j tots iguals a $\{\mu_i + 1, \dots, 2\mu_i\}$.
- \vdots
- σ_i tasques que només es poden assignar a les darreres μ_i màquines, per tant els seus conjunts M_j són tots iguals a $\{(\pi_i - 1)\mu_i + 1, \dots, m\}$.

Per tant tenim per a cada node i π_i grups de σ_i tasques. **Totes** les σ_i tasques de cada grup es poden assignar a **totes** les μ_i màquines corresponents. Aleshores podem repartir aquestes tasques equitativament, de manera que a cada una de les μ_i màquines se li assignen $\lceil \frac{\sigma_i}{\mu_i} \rceil$ o bé $\lfloor \frac{\sigma_i}{\mu_i} \rfloor$. Aquest raonament es pot aplicar a tots els grups de μ_i màquines a l'hora d'assignar-les les σ_i tasques corresponents. Notis també que la decisió de a quines màquines d'un mateix grup se les assignen una tasca més que a les altres és lliure i es pot prendre segons convengui.

Demostració per inducció de la validesa de la distribució

Vegem per inducció sobre n que a l'hora d'assignar les tasques corresponents al node n (havent assignat ja les tasques dels $n - 1$ nodes anteriors), sempre podem aconseguir que el nombre màxim de tasques assignades en total a una màquina és $\lceil \sum_{i=1, \dots, n} \frac{\pi_i \sigma_i}{m} \rceil$. Pel cas $n = 1$, tenim que seguint el raonament anterior el nombre màxim de tasques assignades a una màquina és

$$\left\lceil \frac{\sigma_1}{\mu_1} \right\rceil = \left\lceil \sigma_1 \frac{T_{IBI}}{T t_1} \right\rceil = \left\lceil \sigma_1 \frac{mcm(T_t)}{T_{t_1}} \cdot \frac{T_{IBI}}{mcm(T_t)} \right\rceil = \left\lceil \sigma_1 \frac{\pi_1}{m} \right\rceil$$

Per al cas $n - 1 \Rightarrow n$, aplicam inducció i tenim que per cada grup de μ_x màquines (com que s'han juntat les distribucions de les tasques de diversos nodes, no té perquè ser μ_{n-1}) hi ha a_x màquines amb $\lceil \sum_{i=1, \dots, n-1} \frac{\pi_i \sigma_i}{m} \rceil$ tasques assignades i $b_x = \mu_x - a_x$ màquines amb $\lfloor \sum_{i=1, \dots, n-1} \frac{\pi_i \sigma_i}{m} \rfloor$ tasques. Si distribuïm primer de manera independent les tasques del node n , tenim igualment que per cada μ_n màquines n'hi ha a_n

amb $\lceil \frac{\pi_n \sigma_n}{m} \rceil$ tasques assignades i $b_n = \mu_n - a_n$ màquines amb $\lfloor \frac{\pi_n \sigma_n}{m} \rfloor$ tasques. Si alguna d'aquestes dues fraccions és entera basta afegir els slots del node n a la distribució de slots de $n - 1$ nodes per acabar, ja que aleshores tendríem un màxim de $\lceil \sum_{i=1, \dots, n-1} \frac{\pi_i \sigma_i}{m} \rceil + \lceil \frac{\pi_n \sigma_n}{m} \rceil = \lceil \sum_{i=1, \dots, n} \frac{\pi_i \sigma_i}{m} \rceil$ tasques assignades a cada màquina.

Fita del nombre de màquines amb una tasca extra

El nombre total de tasques assignades en aquest punt és

$$\begin{aligned} & a_x \frac{m}{\mu_x} \left\lceil \sum_{i=1, \dots, n-1} \frac{\pi_i \sigma_i}{m} \right\rceil + b_x \frac{m}{\mu_x} \left\lfloor \sum_{i=1, \dots, n-1} \frac{\pi_i \sigma_i}{m} \right\rfloor + a_n \frac{m}{\mu_n} \left\lceil \frac{\pi_n \sigma_n}{m} \right\rceil + b_n \frac{m}{\mu_n} \left\lfloor \frac{\pi_n \sigma_n}{m} \right\rfloor = \\ & = a_x \frac{m}{\mu_x} \left(\left\lfloor \sum_{i=1, \dots, n-1} \frac{\pi_i \sigma_i}{m} \right\rfloor + 1 \right) + b_x \frac{m}{\mu_x} \left\lfloor \sum_{i=1, \dots, n-1} \frac{\pi_i \sigma_i}{m} \right\rfloor + \\ & \quad + a_n \frac{m}{\mu_n} \left(\left\lfloor \frac{\pi_n \sigma_n}{m} \right\rfloor + 1 \right) + b_n \frac{m}{\mu_n} \left\lfloor \frac{\pi_n \sigma_n}{m} \right\rfloor = \\ & = \frac{m}{\mu_x} \left\lfloor \sum_{i=1, \dots, n-1} \frac{\pi_i \sigma_i}{m} \right\rfloor (a_x + b_x) + a_x \frac{m}{\mu_x} + \frac{m}{\mu_n} \left\lfloor \frac{\pi_n \sigma_n}{m} \right\rfloor (a_n + b_n) + a_n \frac{m}{\mu_n} = \\ & = m \left\lfloor \sum_{i=1, \dots, n-1} \frac{\pi_i \sigma_i}{m} \right\rfloor + m \left\lfloor \frac{\pi_n \sigma_n}{m} \right\rfloor + a_x \frac{m}{\mu_x} + a_n \frac{m}{\mu_n} \end{aligned}$$

Recordem que aquest nombre total de tasques també és igual a $\sum_{i=1, \dots, n} \pi_i \sigma_i$.

Lema per a la suma de parts enters

En aquest punt requerim emprar un lema que ens assegura que, donada l'expressió $\lfloor x \rfloor + \lfloor y \rfloor$, aquesta és igual a $\lfloor x + y \rfloor$ o bé a $\lfloor x + y \rfloor - 1$. En efecte, $\lfloor x \rfloor$ es defineix com el nombre enter que compleix $x - 1 < \lfloor x \rfloor \leq x$, i.e. que pertany a l'interval $(x - 1, x]$. D'aquí tenim que $\lfloor x \rfloor + \lfloor y \rfloor$ és l'enter tal que

$$\begin{aligned} \lfloor x \rfloor + \lfloor y \rfloor & \in (x + \lfloor y \rfloor - 1, x + \lfloor y \rfloor] \Rightarrow \lfloor x \rfloor + \lfloor y \rfloor \in (x + y - 2, x + y] \Rightarrow \\ & \Rightarrow \lfloor x \rfloor + \lfloor y \rfloor \in (x + y - 2, x + y - 1] \vee \lfloor x \rfloor + \lfloor y \rfloor \in (x + y - 1, x + y] \end{aligned}$$

D'on es comprova el lema. Ara, si fos $\lceil \sum_{i=1, \dots, n-1} \frac{\pi_i \sigma_i}{m} \rceil + \lfloor \frac{\pi_n \sigma_n}{m} \rfloor = \lceil \sum_{i=1, \dots, n} \frac{\pi_i \sigma_i}{m} \rceil - 1 \Rightarrow \lceil \sum_{i=1, \dots, n-1} \frac{\pi_i \sigma_i}{m} \rceil + \lceil \frac{\pi_n \sigma_n}{m} \rceil = \lceil \sum_{i=1, \dots, n} \frac{\pi_i \sigma_i}{m} \rceil$, tenim de nou que simplement juntant la distribució dels $n - 1$ nodes amb la del node n obtenim el resultat desitjat. Per tant ens restringim a $\lceil \sum_{i=1, \dots, n-1} \frac{\pi_i \sigma_i}{m} \rceil + \lfloor \frac{\pi_n \sigma_n}{m} \rfloor = \lceil \sum_{i=1, \dots, n} \frac{\pi_i \sigma_i}{m} \rceil$.

Ara suposam per contradicció que la suma de màquines que tenen assignades una tasca més a les dues distribucions és major que el nombre total de màquines, i.e. $a_x \frac{m}{\mu_x} + a_n \frac{m}{\mu_n} > m$. Igualant les dues expressions del nombre total de tasques obtenim

$$\begin{aligned} \sum_{i=1, \dots, n} \pi_i \sigma_i & = m \left\lfloor \sum_{i=1, \dots, n-1} \frac{\pi_i \sigma_i}{m} \right\rfloor + m \left\lfloor \frac{\pi_n \sigma_n}{m} \right\rfloor + a_x \frac{m}{\mu_x} + a_n \frac{m}{\mu_n} \Rightarrow \\ \Rightarrow \frac{\sum_{i=1, \dots, n} \pi_i \sigma_i}{m} & = \left\lfloor \frac{\sum_{i=1, \dots, n-1} \pi_i \sigma_i}{m} \right\rfloor + a_x \frac{1}{\mu_x} + a_n \frac{1}{\mu_n} > \left\lfloor \frac{\sum_{i=1, \dots, n} \pi_i \sigma_i}{m} \right\rfloor + 1 > \frac{\sum_{i=1, \dots, n} \pi_i \sigma_i}{m} \end{aligned}$$

I arribam a contradicció.

Distribució final

Com que s'ha vist que la suma del nombre de màquines amb una tasca de més assignada no pot ser major que m , basta triar les màquines amb un tasca de més a la distribució del node n de manera que no es repeteixin amb les que tenen una tasca de més a la distribució dels $n - 1$ nodes anteriors.

Dividim les màquines del model paral·lelament en grups de μ_x màquines i en grups de μ_n màquines. Es tracta d'assignar una tasca extra a les màquines de cada grup de μ_n (les de la distribució del node n) de manera que "encaixin" amb les dels grup de μ_x màquines que no en tenen (les de la distribució dels $n - 1$ nodes anteriors). Assumim sense pèrdua de generalitat que $\mu_x \geq \mu_n$. Assignam les tasques extra segons aquest procés:

- Prenem el primer grup de μ_n màquines i assignam una tasca extra a les màquines del primer grup de μ_x màquines que estan lliures.
- Si no queden tasques lliures suficients prenem les tasques lliures del següent grup de μ_x màquines que també pertanyen al primer grup de μ_n màquines.
- Si cal, podem reassignar les tasques extra de la distribució dels primers $n - 1$ nodes dins el grup de μ_x màquines de manera que quedin lliures les màquines que pertanyen al grup de μ_n màquines.

Si seguint aquestes passes recorrem $\mu^* = mcm(\mu_x, \mu_n)$ quedant encara una tasca extra que no es pugui assignar a una màquina lliure, tendrem que la suma de les primeres μ^* màquines amb una tasca extra a alguna de les dues distribucions és superior a μ^* . Com que el nombre de tasques extra de la corresponent distribució és igual a cada grup de μ_x i μ_n màquines és igual, també ho és en total a cada grup de μ^* màquines. Això vol dir que el nombre total de tasques extra és superior a m , cas que no es pot donar.

Aconseguim per tant una distribució final on cada màquina té assignades com a màxim $\lceil \sum_{i=1, \dots, n} \frac{\pi_i \sigma_i}{m} \rceil$ tasques.

Cal tenir en compte que aquesta distribució dels slots de transmissió té un inconvenient. Això és degut a què els slots de dades d'un node es reparteixen entre tots els intervals IBI que conformen el seu període de transmissió. És preferible que un node transmeti totes les dades del seu període de transmissió en un mateix IBI, ja que així s'estalvia l'energia que suposa passar d'estat inactiu a actiu i viceversa. Fer transmetre a un node en més intervals IBI dels que cal pot resultar ineficient. Per tant, en la pràctica, un cop hem obtingut la distribució de slots que minimitza T_{SAP} , l'ideal seria permutar els slots de transmissió de manera que els slots de dades d'un mateix node estiguin junts, evitant augmentar el nombre de slots d'un mateix IBI. Això sí, sempre s'ha d'assegurar que el nombre de slots d'un node dins el seu període de transmissió és el requerit per enviar les dades pendents.

4.3.2 Exemple de resolució

Per acabar explicam com s'aplicaria el procés descrit per trobar una distribució de tasques òptima en un exemple concret. Consideram una xarxa de quatre nodes amb

4. TEORIA DE SCHEDULING

períodes de transmissió $T_t = (300, 600, 600, 1200)[ms]$ i nombre de slots de dades a transmetre per període $\sigma = (1, 3, 3, 3)$, respectivament. El T_{IBI} és de $300ms$. D'aquí obtenim un nombre total de màquines $m = \frac{mcm(T_t)}{T_{IBI}} = \frac{1200}{300} = 4$, i un nombre de tasques $\tau = \sum_{i=1, \dots, 4} \pi_i \sigma_i = 4 \cdot 1 + 2 \cdot 3 + 2 \cdot 3 + 1 \cdot 3 = 19$. L'objectiu serà obtenir una distribució amb un màxim de $\lceil \frac{\tau}{m} \rceil = 5$ tasques per màquina. Anam assignant totes les tasques corresponents a les transmissions d'un sensor rere l'altre. El procés s'explica en les següents passes, i s'inclou una explicació visual a la figura 4.2.

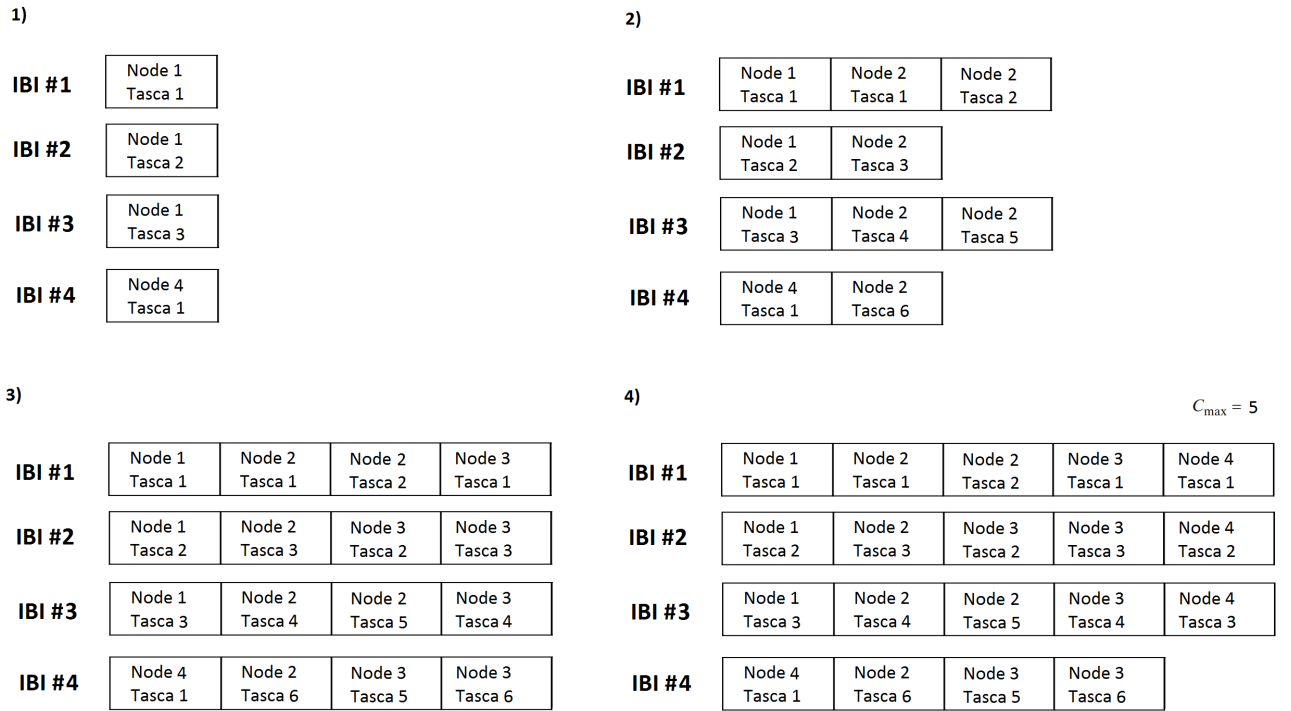


Figura 4.2: Procediment per trobar la distribució òptima a l'exemple

1. El node 1 transmet $\sigma_1 = 1$ slots cada $\mu_1 = \frac{T_{t1}}{T_{IBI}} = 1$ IBIs. Això defineix $\pi_1 \sigma_1$ tasques on cada una només es pot processar en una única màquina, i.e. $M_{1,1} = \{1\}$, $M_{1,2} = \{2\}$, $M_{1,3} = \{3\}$ i $M_{1,4} = \{4\}$. Com s'ha dit, la idea és assignar a cada màquina el mateix nombre de tasques, encara que en aquest cas no tenim altra opció. Per tant assignam cada tasca a l'única màquina que la pot processar.
2. El node 2 transmet 3 slots cada 2 IBIs. Això defineix sis tasques diferents. Tres d'elles es poden assignar a les dues primeres màquines; i les altres tres, a les dues darreres màquines, i.e. $M_{2,1} = M_{2,2} = M_{2,3} = \{1, 2\}$ i $M_{2,4} = M_{2,5} = M_{2,6} = \{3, 4\}$. Com que no podem tenir el mateix nombre de tasques a totes les màquines hi haurà d'haver dues màquines amb dues tasques assignades (la tasca "extra") i dues màquines amb una tasca assignada. Assignam arbitràriament la tasca extra a la primera màquina dels dos conjunts.

3. Al node 3 es repeteix el mateix cas que a la passa anterior, amb la diferència que ara no totes les màquines tenen el mateix nombre de tasques. Tenim sis tasques amb $M_{3,1} = M_{3,2} = M_{3,3} = \{1, 2\}$ i $M_{3,4} = M_{3,5} = M_{3,6} = \{3, 4\}$. Hem d'aconseguir assignar les tasques en aquesta passa de manera que el màxim de tasques assignades a cada màquina sigui $\lceil \sum_{i=1,2,3} \frac{\pi_i \sigma_i}{4} \rceil = \lceil \frac{16}{4} \rceil = 4$. Per això, hem d'encaixar les màquines a les que les assignem una tasca extra amb les que no tenien una tasca extra abans. Les màquines que abans ja tenien una tasca més són 1 i 3, per tant assignam dues tasques a les màquines 2 i 4, i només una a les màquines 1 i 3.
4. Finalment, el node 4 transmet 3 slots cada 4 IBIs, que seran tres tasques amb $M_{4,1} = M_{4,2} = M_{4,3} = \{1, 2, 3, 4\}$. Per assignar les tasques de la manera més equitativament possible, assignam una tasca a cada una de les tres primeres màquines. Assolim l'objectiu de tenir un màxim de 5 tasques per cada màquina.

Com s'ha dit, és desitjable que un sensor transmeti tots els seus slots en un mateix IBI sempre que sigui possible. Les tasques a la distribució obtinguda es poden permutar de manera que aquesta condició es compleixi per alguns sensors, mentre que les restriccions d'assignació de màquines i el màxim nombre de tasques per màquina se segueixen complint. Aplicant aquestes permutacions, la distribució quedaria com es pot veure a la figura 4.3.

IBI #1	Node 1 Tasca 1	Node 2 Tasca 1	Node 2 Tasca 2	Node 2 Tasca 3	Node 4 Tasca 1
IBI #2	Node 1 Tasca 2	Node 3 Tasca 1	Node 3 Tasca 2	Node 3 Tasca 3	Node 4 Tasca 2
IBI #3	Node 1 Tasca 3	Node 2 Tasca 4	Node 2 Tasca 5	Node 2 Tasca 6	Node 4 Tasca 3
IBI #4	Node 4 Tasca 1	Node 3 Tasca 4	Node 3 Tasca 5	Node 3 Tasca 6	

Figura 4.3: Permutacions a la distribució per agrupar els slots dels sensors 2 i 3

ANÀLISI DE RESULTATS

L'objectiu d'aquest capítol es tracta d'analitzar el comportament dels algoritmes proposats i el resultat del seu possible ús en una xarxa SmartBAN. Més concretament, ens interessa veure com afecta l'ús d'aquests algoritmes al consum d'energia dels sensors que formen el conjunt de nodes de la xarxa. Aquesta anàlisi es realitzarà simulant el comportament de la xarxa amb MATLAB. Majoritàriament s'estudien els casos analitzats a [5] i com canvien en cas d'aplicar l'optimització del SAP desenvolupada al capítol anterior.

5.1 Rendiment energètic i temps de bateria dels sensors

Donades les característiques dels sensors que actuen com a nodes a una xarxa SmartBAN, el seu estalvi d'energia és possiblement el paràmetre més important a estudiar. Estudiam el possible estalvi energètic obtingut amb de la utilització de l'eeAlgorithm a partir del rendiment energètic i temps de bateria resultants. Primer de tot requerim d'expressions analítiques per aquests paràmetres.

Expressió analítica per al rendiment energètic

Obtenim el rendiment energètic a partir de la fracció entre la potència que gastaria el sensor en cas d'estar actiu tot el temps, i la potència que gasta realment. Per això cal considerar els paràmetres P_{ACT} , la potència que gasta un sensor quan està actiu; P_{SLP} , la potència que gasta un sensor quan està en repòs; i P_{SW} , la potència que gasta un sensor per passar d'un d'aquests dos estats a l'altre. De la mateixa manera, consideram els paràmetres que indiquen el temps que passa un sensor i en cada un d'aquests estats durant el seu període de transmissió T_{t_i} : T_{ACT_i} , T_{SLP_i} i T_{SW_i} . Notis que consideram que les potències no varien entre els distints sensors (suposam que són el mateix tipus d'aparell), però els paràmetres de temps sí. A continuació es calculen cadascun d'aquests intervals de temps.

- Un node estarà actiu durant la transmissió per part del hub de la D-Beacon i durant la transmissió dels seus slots de dades. La transmissió de la D-Beacon té una durada $T_{Dbcn} = 0.224ms$. Un node i escolta tantes D-Beacon en el seu període de transmissió com intervals IBI aquest conté, que seran en total $\mu_i = \frac{T_{t_i}}{T_{IBI}}$. El nombre de slots de dades que envia un node en aquest període era $\sigma_i = \frac{T_{t_i}}{T_{g_i}} I_{S_i}$. D'aquí obtenim el seu període actiu com $T_{ACT_i} = \mu_i T_{Dbcn} + \sigma_i T_S$ (on recordem que T_S és la durada d'un slot de dades).
- Partint de la suposició de què un node qualsevol tarda $T_{sw} = 6.25\mu s$ en canviar d'estar actiu a estar en repòs i viceversa, falta trobar el nombre de vegades que aquest canvia d'estat en un període de transmissió. Sigui ν_i el nombre d'intervals IBI en què el node i transmet dins el seu període de transmissió. En la versió de l'eeAlgorithm aplicada originalment a [5], es considerava sempre que $\nu_i = 1$. Però si partim d'una distribució de slots obtinguda a partir del procés de minimització de T_{SAP} , aquest nombre pot variar entre 1 i μ_i . En qualsevol cas, un node canviarà d'estat dues vegades a cada trama D-Beacon (una per activar-se i l'altra per tornar a passar a repòs), i dues vegades també per cada IBI on hagi de transmetre. Per tant tenim $T_{SW_i} = 2(\mu_i + \nu_i) T_{sw}$.
- Finalment obtenim el temps de repòs del node i com $T_{SLP_i} = T_{t_i} - T_{ACT_i} - T_{SW_i}$

D'aquesta manera calculam el rendiment energètic del node i com

$$\eta_i = 1 - \frac{T_{ACT_i} P_{ACT} + T_{SW_i} P_{SW} + T_{SLP_i} P_{SLP}}{T_{ACT_i} P_{ACT}} \quad (5.1)$$

Els valors de les potències els obtenim a partir de [6] com $P_{ACT} = 136mW$, $P_{SW} = 1mW$ i $P_{SLP} = 0.2mW$.

Temps de bateria

Calculam el temps de bateria dels sensors a partir de l'energia que consumeixen. Per això consideram la intensitat dels sensors en cinc estats diferents: quan el sensor està rebent dades, transmetent dades, en espera (actiu però sense rebre ni transmetre dades), despertant-se (passant d'estar en repòs a actiu) i en repòs. Els valors d'aquestes intensitats ens són proporcionats per [8], i són $I_{rx} = 13mA$, $I_{tx} = 5.5mA$, $I_{wait} = 7mA$, $I_{wup} = 7mA$, i $I_{slp} = 23\mu A$, respectivament. Ara calculam el consum d'energia del sensor i a cada un d'aquests estats en un dels seus períodes de transmissió.

- El sensor i es troba rebent dades cada D-Beacon, i com hem vist això succeeix μ_i vegades a cada període de transmissió. Per tant obtenim un consum d'energia igual a $\mu_i \cdot T_{Dbcn} \cdot I_{rx}$.
- Recordem que cada slot de transmissió es componia de quatre períodes: a un d'ells el sensor transmet les dades, a un altre rep un ACK, i als altres dos està en espera (períodes Inter Frame Space (IFS)). D'aquesta manera obtenim l'energia consumida pel sensor i en els seus σ_i slots de transmissió com $\sigma_i (T_{Data} \cdot I_{tx} + T_{ACK} \cdot I_{rx} + 2T_{IFS})$. T_{ACK} i T_{IFS} s'especifiquen en el modelat de la xarxa (3.1), i tenim que $T_{Data} = T_S - T_{ACK} - T_{IFS}$.

- Havíem calculat a l'apartat anterior el temps T_{SW_i} que passa el node i canviant d'estat actiu a repòs i viceversa. Si consideram només el temps que passa despertant-se, obtenim aquest consum de bateria com $\frac{T_{SW_i}}{2} \cdot I_{wup}$.
- Ens queda considerar el temps que passa el sensor en repòs, o canviant d'estat actiu a repòs, on la intensitat és la mateixa. Recuperant també de l'apartat anterior el temps T_{SLP_i} que passa el node en repòs, obtenim el restant consum d'energia com $(T_{SLP_i} + \frac{T_{SW_i}}{2}) \cdot I_{slp}$

Sumant aquests elements obtenim el consum total de bateria com

$$Q_{TOTAL_i} = \mu_i \cdot T_{Dbcn} \cdot I_{rx} + \sigma_i (T_{Data} \cdot I_{tx} + T_{ACK} \cdot I_{rx} + 2T_{IFS}) + \frac{T_{SW_i}}{2} \cdot I_{wup} + (T_{SLP_i} + \frac{T_{SW_i}}{2}) \cdot I_{slp} \quad (5.2)$$

Per tant obtenim el temps de bateria del sensor i com

$$T_{BAT_i} = \frac{Q_{BAT}}{Q_{TOTAL_i}} T_i \quad (5.3)$$

On $Q_{BAT} = 30mA \cdot h$ és la capacitat de la bateria dels sensors (valor obtingut de [8]).

5.2 Simulació de la xarxa

Volem simular el comportament d'una xarxa SmartBAN que faci ús de l'eeAlgoritme i avaluar-ne el rendiment i el temps de bateria dels sensors. Consideram dos casos distints de xarxes per simular:

- Una xarxa de sis sensors amb períodes de generació de dades $T_g = (100, 100, 100, 200, 500, 1000) [ms]$
- Una xarxa de vuit sensors amb períodes de generació de dades $T_g = (100, 100, 100, 200, 250, 500, 750, 1000) [ms]$

A ambdós casos i per a tots els sensors, el volum de les dades generades és $V = 1600bits$, el retard màxim permès de les dades és $D = 3000ms$, i el retard màxim per emergències és $D_E = 200ms$. Aquests exemples de xarxes provenen de [6] i són també els estudiats a [5]. A les figures 5.1 i 5.2 es veuen els resultats obtinguts pel rendiment energètic i el temps de bateria dels sensors a les dues xarxes.

A la xarxa de sis sensors s'obté un $T_{IBI} = 2000ms$ i un període de transmissió $T_t = 2000ms$ per a tots els sensors. En el cas de vuit sensors tenim $T_{IBI} = 1500ms$ i períodes de transmissió $T_t = (1500, 1500, 1500, 3000, 1500, 1500, 1500, 3000) [ms]$. A les dues xarxes T_{IBI} no varia en cas d'aplicar o no optimització del T_{SAP} , per tant podem emprar les distribucions de slots obtingudes amb l'eeAlgorithm original, on cada node transmet en un únic interval IBI dins el seu període de transmissió, i.e. $v_i = 1$ per a tots els sensors.

Veim en general que els períodes de transmissió i T_{IBI} són relativament grans, bastant pròxims al retard màxim permès D . Això és degut principalment a què les condicions de les xarxes són poc restrictives. El volum de dades que envien els sensors no és massa gran, i no consideram per exemple la necessitat de repetició de trames en cas de que hi hagi renou a la xarxa. L'estudi d'aquests casos més restrictius es realitza en els següents apartats.

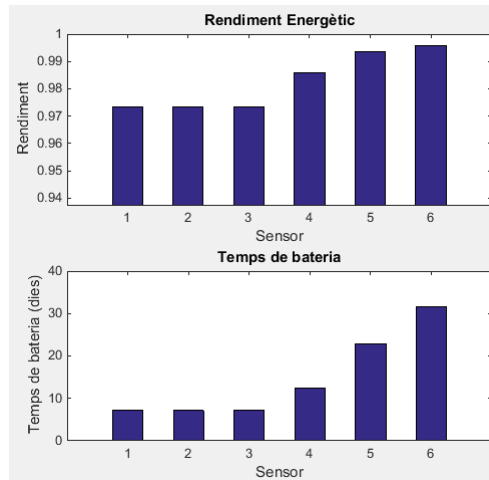


Figura 5.1: Rendiment i temps de bateria per a la xarxa de 6 sensors

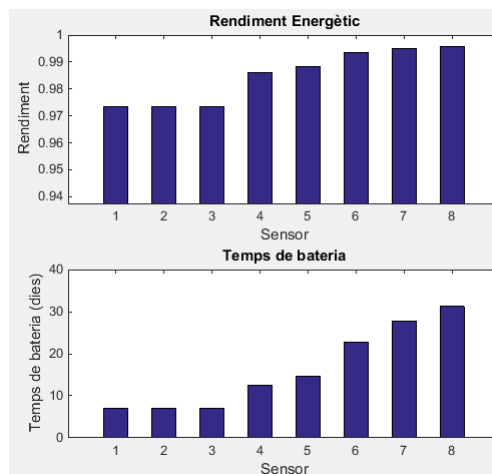


Figura 5.2: Rendiment i temps de bateria per a la xarxa de 8 sensors

5.3 Modes de transmissió

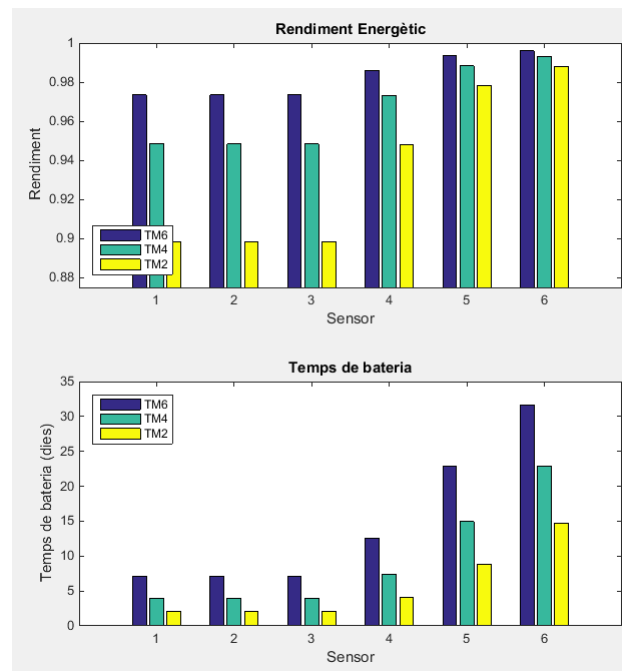
Recuperant el que s'ha vist a la explicació de l'estàndard ETSI ([3], [4]) al capítol 2, amb l'objectiu de millorar la probabilitat d'error a l'hora de transmetre, es pot realitzar la transmissió de la PPDU un nombre $N_{REP} \in \{1, 2, 4\}$ de repeticions. L'estàndard també ofereix la possibilitat d'ús de codificació BCH. Si combinem aquest nombre variable de repeticions amb la utilització opcional de BCH, obtenim 6 maneres diferents de transmetre la PPDU que podem variar segons els requeriments i les característiques de la xarxa. Aquestes combinacions les anomenem modes de transmissió (TM), i s'enuncien a la taula 5.1. A l'apartat anterior no consideràvem repeticions de la PPDU ni utilització de BCH, de manera que empràvem el que anomenàriem mode de transmissió 6 (TM6).

Mode de transmissió	6	5	4	3	2	1
N_{REP}	1	1	2	2	4	4
Codificació BCH	No	Sí	No	Sí	No	Sí

Taula 5.1: Descripció dels sis modes de transmissió

5.3.1 Variació en el nombre de repeticions

Vegem una comparativa del rendiment i del temps de bateria dels sensors variant el valor de N_{REP} (TM6, TM4 i TM2). Empram el mateix exemple de xarxa de sis sensors de l'apartat anterior. El resultat es veu a la figura 5.3.

Figura 5.3: Comparativa de resultats segons N_{REP}

Els valors de T_{IBI} obtinguts per els tres modes són $2000ms$, $1000ms$ i $500ms$. El cas de TM2 és l'únic on tenim una millora del T_{IBI} aplicant optimització del T_{SAP} : amb la versió original de l'eeAlgorithm resulta $400ms$. Amb un $T_{IBI} = 500ms$ els períodes de transmissió són $T_t = \{500, 500, 500, 1000, 500, 1000\}[ms]$, i la distribució de slots obtinguda que minimitza T_{SAP} en aquest cas es veu a la figura 5.4.

5.3.2 Millora en l'ús de minimització de T_{SAP}

La millora del T_{IBI} en el cas de TM2 és deguda a la distribució subòptima dels slots a l'eeAlgorithm original. A la distribució de la versió original de l'algoritme per $T_{IBI} = 500ms$, els slots del sensor 4 es transmetrien tots al mateix IBI. Això implica que com a mínim hi hauria un SAP amb almenys 20 slots. Com que en aquest cas la durada dels slots és $T_S = 10ms$, tendríem que $T_{SAP} = 200ms = D_E$, i per tant no es complirien els requisits del retard per el tràfic d'emergència, el que implicaria una reducció de T_{IBI} a

5. ANÀLISI DE RESULTATS

IBI #1	Sensor 1 5 slots	Sensor 2 5 slots	Sensor 3 5 slots	Sensor 4 3 slots	Sensor 5 1 slot	
						$C_{\max} = 19$ slots
IBI #2	Sensor 1 5 slots	Sensor 2 5 slots	Sensor 3 5 slots	Sensor 4 2 slots	Sensor 5 1 slot	Sensor 6 1 slot

Figura 5.4: Distribució òptima de slots per $T_{IBI} = 500ms$

l'algoritme. En canvi, amb la distribució òptima, tenim $T_{SAP} = 190ms < D_E$, i per tant el T_{IBI} de $500ms$ és vàlid.

Podem aprofitar per estudiar com milloren el rendiment i el temps de bateria dels sensors aplicant minimització del T_{SAP} . La comparativa es presenta a la figura 5.5. Observant la gràfica podem dir que el fet d'augmentar el T_{IBI} gràcies a la minimització del T_{SAP} millora lleugerament el consum d'energia dels sensors. D'altra banda, l'energia addicional que consumiria el sensor 4 per canviar entre estats no pareix que afecti gaire (el seu temps de bateria augmenta uns 40 minuts emprant minimització del T_{SAP}).

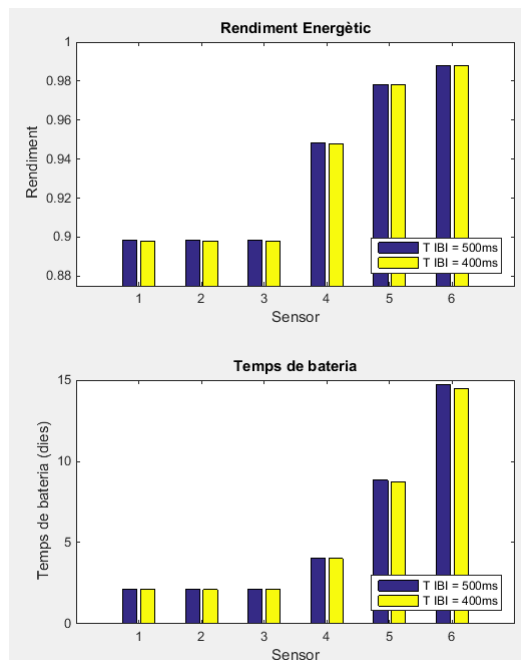


Figura 5.5: Comparativa de l'ús de minimització de T_{SAP}

5.3.3 Ús de BCH

Per acabar la secció, estudiem l'efecte que té en el consum d'energia dels sensors l'ús de BCH. Considerem la mateixa xarxa de sis sensors, però aquest cop el volum de les dades generades és $V = 400$ bits. Així, el fet d'afegir bits de codificació BCH sobre un paquet més petit influirà més. Per fer la comparativa, obtenim els resultats del rendiment energètic i del temps de bateria dels sensors en els modes de transmissió TM2 (sense BCH) i TM1 (amb BCH). Els resultats es poden veure a la figura 5.6.

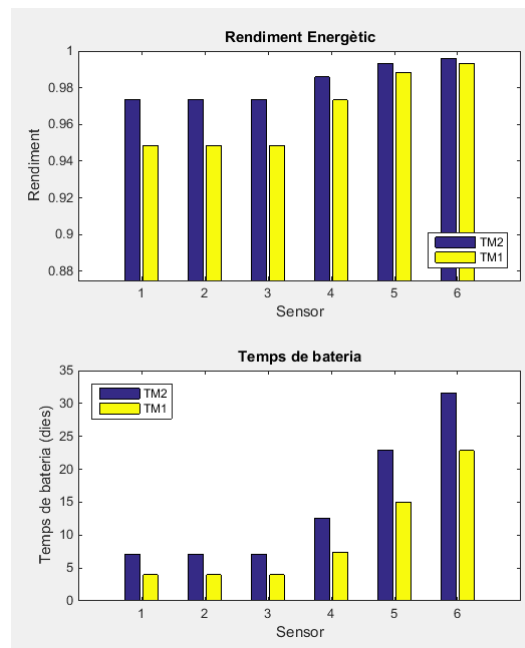


Figura 5.6: Comparativa de l'ús de codificació BCH

Donat l'increment de bits a transmetre com a conseqüència de l'ús de BCH, això provoca que els sensors consumeixin més energia, resultant en una disminució del rendiment i del temps de bateria. En aquests casos el T_{IBI} obtingut és de $2000ms$ pel TM2 i de $1000ms$ pel TM1, de manera que torna a coincidir amb la versió original de l'Algorithm sense optimització del T_{SAP} .

5.4 Algorisme de selecció de mode de transmissió

En aquesta secció es pretén estudiar l'algorisme de selecció pels diferents modes de transmissió proposat a [5]. L'objectiu d'aquest algorisme és emprar sempre el mode de transmissió ideal a partir d'una certa relació senyal-renou E_b/N_0 i d'una probabilitat d'error objectiu P_0 . Les transmissions s'haurien de realitzar en el TM de major ordre que garanteixi una probabilitat d'error P_0 , donada la relació senyal-renou E_b/N_0 . Això permet que la xarxa compleixi els requisits d'èxit en les transmissions, però sense que els sensors tinguin un consum d'energia major del necessari.

Per a la implementació de l'algorisme només necessitam, donada una P_0 objectiu fixa, el conjunt de llindars $\Gamma = \{\gamma_1, \gamma_2, \dots, \gamma_6\}$ de E_b/N_0 a partir dels quals es canvia de

mode de transmissió. Per exemple, si el valor de E_b/N_0 es troba entre γ_3 i γ_4 voldrà dir que el mode seleccionat és TM3. Si E_b/N_0 és major que γ_6 , el mode seleccionat és TM6; i si és menor que γ_1 , vol dir que la P_0 objectiu no es pot assolir (en aquest cas no hi hauria transmissió).

A [5] hi trobam també un model de la relació entre E_b/N_0 i P_0 suposant un renou blanc gaussià a la xarxa. A partir d'aquest model, els llindars del conjunt Γ s'obtenen com:

$$\gamma_i = \frac{1}{g_i} \ln\left(\frac{a_i}{P_0}\right) \quad \forall i \in \{1, \dots, 6\} \quad (5.4)$$

On els valors dels distints a_n i g_n s'obtenen a partir del mateix model.

Ara passam a estudiar com afecta aquest petit algoritme al rendiment i al temps de bateria dels sensors. Consideram un exemple de xarxa de 7 sensors on els sis primers sensors transmeten cada un en un TM distint. Els períodes de generació de dades i el TM de cada sensor es poden veure a la taula 5.2. El retard màxim per a les dades generades de tots els sensors és novament $D = 3s$, i el retard per a les dades d'emergència és $D_E = 200ms$. Per el volum de les dades que generen els sensors consideram tres casos distints: $V = 50$ Bytes, $V = 200$ Bytes i $V = 500$ Bytes.

Sensor	1	2	3	4	5	6	7
T_{g_i} [ms]	100	100	200	250	500	750	1000
TM	1	2	3	4	5	6	-

Taula 5.2: Períodes de generació i TM seleccionat dels set sensors

Volem estudiar el comportament de l'algoritme de selecció de TM sobre el sensor 7. Consideram primer una probabilitat d'error objectiu $P_0 = 0.01$ i estudiem el TM seleccionat i el temps de bateria obtingut segons el valor de E_b/N_0 i per els tres valors proposats de V . Després feim el mateix fixant un $E_b/N_0 = 7dB$ i obtenint el TM seleccionat i el temps de bateria segons P_0 , també per els tres casos del valor de V . Els resultats es veuen a les figures 5.7 i 5.8, respectivament.

A les gràfiques es veu que, com era d'esperar, quan augmenta E_b/N_0 , és més fàcil que les transmissions es realitzin amb èxit, i per tant es redueix la necessitat de l'ús de BCH o de repetició de PPDU, donant lloc a la selecció de TMs d'ordre més elevat. Això implica una reducció del consum d'energia del sensor, donat que pot transmetre les seves dades afegint menys redundància, i per tant el temps de bateria també millora. De la mateixa manera, una major P_0 implica una menor exigència per a l'èxit de les transmissions, el que també resulta en l'ús de TMs d'ordre més elevat i major temps de bateria. Observam també que un major volum de dades generades resulta en la necessitat d'un millor E_b/N_0 o d'una P_0 menys estricta per poder transmetre en TMs d'ordre major, el que fa que el temps de bateria es redueixi.

5.4. Algorisme de selecció de mode de transmissió

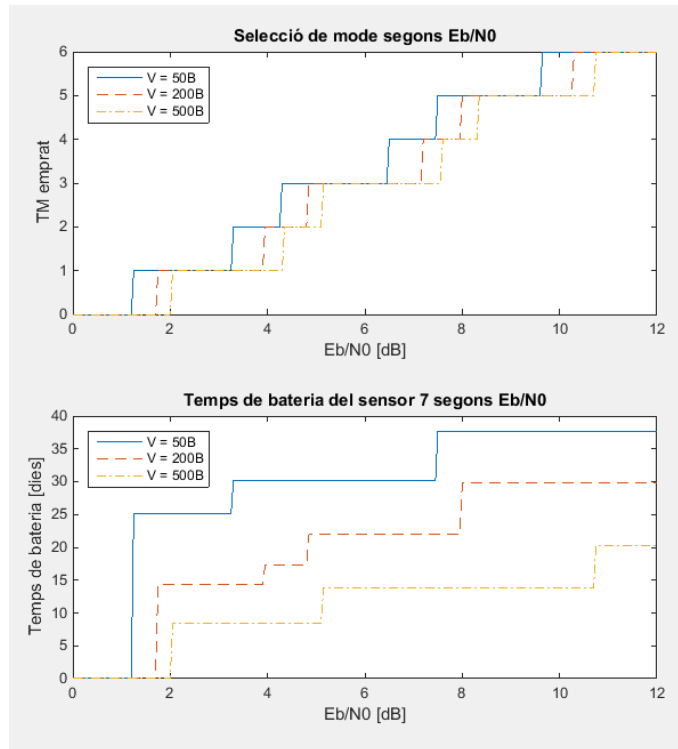


Figura 5.7: TM seleccionat i temps de bateria segons E_b/N_0

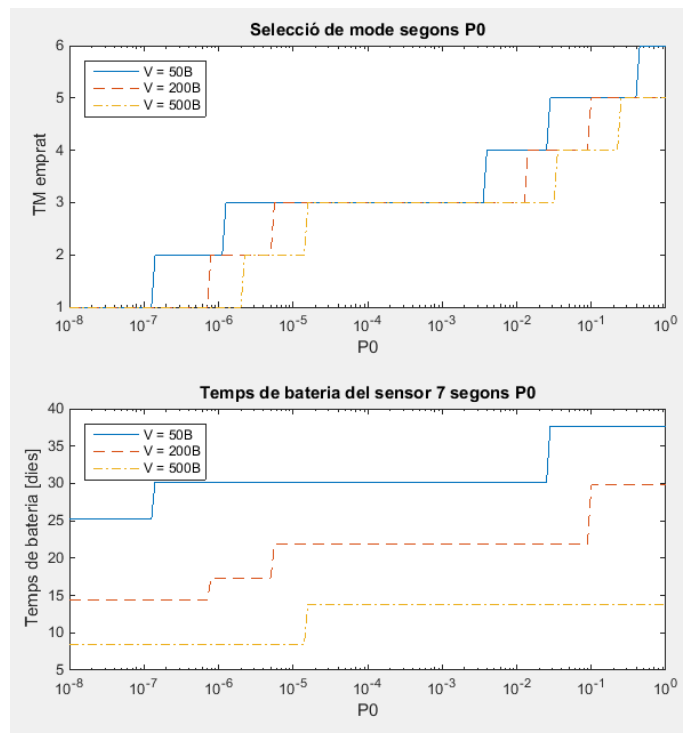


Figura 5.8: TM seleccionat i temps de bateria segons P_0

CONCLUSIONS

En aquest treball s'ha fet un estudi de las xarxes SmartBAN proposades per a la monitorització continuada de pacients crònics com a solució al problema d'abastiment de recursos en l'àmbit sanitari. Amb l'objectiu de minimitzar el consum d'energia dels dispositius delicats que componen aquestes xarxes, s'ha introduït l'estàndard ETSI i s'ha explicat el seu funcionament.

També s'ha introduït l'eeAlgorithm proposat a [5] com un algoritme de planificació per a xarxes SmartBAN. Aquest algoritme aconseguix un augment considerable sobre la durada dels intervals IBI respecte d'altres propostes, el que suposa un augment en el rendiment energètic dels sensors. Tot i així, aquest algoritme no especifica com distribuir els slots de transmissió dins el període d'accés planificat, el que fa que l'algoritme doni peu a millora.

L'aportació principal d'aquest treball ha consistit en millorar l'eeAlgorithm trobant una distribució de slots que minimitza la durada del SAP. Això s'ha aconseguit plantejant un model de *Scheduling* per a solucionar el problema. D'aquesta manera, ha quedat demostrada l'existència d'una distribució dels slots de transmissió dels sensors amb la que s'obté un T_{SAP} òptim. Aquest T_{SAP} és el mateix que s'obtindria en cas de què fos possible una distribució on es repartissin equitativament els slots de transmissió entre els distints intervals IBI. Gràcies a la minimització del T_{SAP} s'ha aconseguit augmentar el T_{IBI} en els casos on la distribució de slots a l'eeAlgorithm no complia els requeriments del retard per les dades d'emergència.

El rendiment dels sensors resultant de la distribució obtinguda sempre és igual o millor que el de la versió original de l'eeAlgorithm. Tot i així, aquesta distribució introdueix una nova problemàtica en què els sensors separen les seves transmissions entre els distints IBI que componen el seu període de transmissió. D'aquesta manera, el rendiment dels sensors encara seria millorable juntant les transmissions d'un mateix sensor en un mateix IBI, per tal d'evitar que aquest s'activi i desactivi més vegades de les necessàries. Queda pendent la implementació d'un algoritme que trobi explícitament una distribució de slots amb el T_{SAP} òptim i on un sensor transmeti totes les dades pendents a un mateix IBI sempre que sigui possible.

6. CONCLUSIONS

Finalment, s'ha pogut visualitzar el consum d'energia dels sensors obtingut a partir d'aquest algoritme, avaluant el seu rendiment energètic i temps de bateria. S'han comparat aquests paràmetres en distintes condicions sobre la xarxa. Quan hi ha renou a la xarxa o la probabilitat d'error desitjada és exigent, es requereix augmentar el volum de les dades enviades mitjançant repeticions de trama o codificació BCH. Això òbviament provoca que el rendiment i el temps de bateria dels sensors disminueixi, a canvi de permetre que se satisfacin els requeriments d'èxit en les transmissions. També s'ha pogut observar com millora el comportament de la xarxa quan s'aplica la distribució de slots òptima respecte de la versió original de l'eeAlgorithm. S'ha vist que l'augment del T_{IBI} aconseguit compensa l'energia consumida pels sensors que en conseqüència s'han d'activar amb més freqüència.

BIBLIOGRAFIA

- [1] J. Lloret, A. Canovas, S. Sendra, and L. Parra, “A smart communication architecture for ambient assisted living,” *IEEE Communications Magazine*, Gener 2015. 1
- [2] P. J. Soh, G. A. E. Vandenbosch, M. Mercuri, and D. M. M. P. Schreurs, “Wearable wireless health monitoring: Current developments, challenges, and future trends,” *IEEE Microware Magazine*, pp. 55 – 70, Maig 2015. 1
- [3] “Smart body area network (smartban); enhanced ultralow power physical layer,” *ETSI TS 103 326 v1.1.1*. 1, 5.3
- [4] “Smart body area network (smartban); low complexity medium access control (mac) for smartban,” *ETSI TS 103 325 v1.1.1*. 1, 5.3
- [5] J. Ramis-Bibiloni and L. Carrasco-Martorell, “Energy-efficient and qos-aware link adaptation scheme and resource allocation algorithm for periodical monitoring traffic in smartbans,” *2019 IEEE Biomedical Circuits and Systems Conference (BioCAS)*, pp. 1–10. 1, 3.2, 5, 5.1, 5.2, 5.4, 6
- [6] L. Ruan, M. Dias, and E. Wong, “Smartban with periodic monitoring traffic: A performance study on low delay and high energy efficiency,” *IEEE Journal of Biomedical and Health Informatics*, vol. 22, no. 2, pp. 471–482, Març 2018. 3.2, 5.1, 5.2
- [7] M. L. Pinedo, *Scheduling. Theory, Algorithms, and Systems*. Springer, 2016. 4, 4.1, 4.3
- [8] R. Matsuo and T. Nabetani, “Performance of simple and smart phy/mac mechanisms for body area networks,” *IEEE International Conference on Communications (ICC)*, pp. 501–506, 2015. 5.1, 5.1