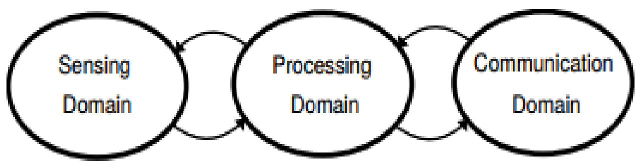
5.1.4 Transitions between domains

The processing domain is formed by a microcontroller, which has a microprocessor responsible to control all the components in the sensor node (ZHENG; JAMALIPOUR, 2009). In this manner, the components that form the sensing and communication domain are peripherals of the microcontroller. Thus, related to the communication between domains, the processing domain works as a master and the sensing and communication domain are slaves (RUSSELL, 2010). This means that there is not direct communication between the sensing and communication domain, and as a consequence, there are not transitions between both domains states. Figure 14 depicts the possible domain transitions.

Figure 14: Domain transitions behavior



Source: Author

According to Figure 14, the energy consumption by domain transitions can be calculated as shown in Equation 5.10. The terms $E_{CPU-Com}$, $E_{Com-CPU}$, $E_{CPU-Sens}$ and $E_{Sens-CPU}$ are the energy consumption by the transitions between: processing to communication, communication to processing, processing to sensing, and sensing to processing, respectively.

$$E_{DT} = E_{CPU-Com} + E_{Com-CPU} + E_{CPU-Sens} + E_{Sens-CPU} \tag{5.10}$$

To calculate the energy consumption for a certain period, it is necessary to count the number of transitions between domains in this period. Then, the number of transitions of each type of domain-transition is multiplied by the average domain-transition

consumption. Finally, the energy consumption is calculated as shown in Equation 5.11. $N_{CPU-Com}$, $N_{Com-CPU}$, $N_{CPU-Sens}$, and $NSens-CPU$ are the number of transitions from processing to communication, communication to processing, processing to sensing, and sensing to processing, respectively. $E_{CPU-Com}$, $E_{Com-CPU}$, $ECPU-Sens$, and $ESens-CPU$ are the average energy consumption from processing to communication, communication to processing, processing to sensing and sensing to processing, respectively.

$$E_{DT} = N_{CPU-Com}E_{CPU-Com} + N_{Com-CPU}E_{Com-CPU} + N_{CPU-Sens}E_{CPU-Sens} + N_{Sens-CPU}E_{Sens-CPU} \tag{5.11}$$

5.1.5 Model considerations

From the analysis conducted in Section 3, it was identified the main elements considered in the literature to model the energy consumption, but also it was identified

5.2 Energy consumption prediction with Markov chain

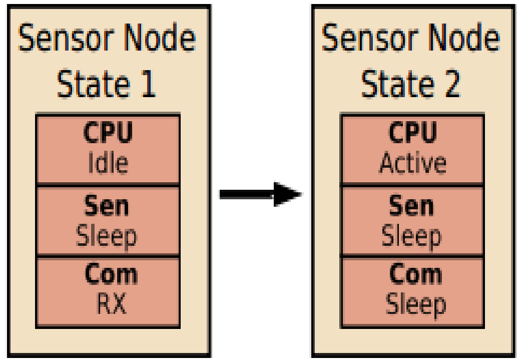
In Section 5.1, the energy consumption was modeled as the sum of the consumption of three domains and the cost of transitions between domains. This approach can give the impression that the sensor node is constantly jumping among all the states,

being in only one state at the same time. But, what really happens is that the node state is a combination of three states, one state of each domain.

To predict the energy consumption with the Markov chain approach, based on the energy model proposed, the first step is to determine which are the combinations of states that represents typical sensor node behavior. Then, each combination is defined as a *sensor node state*, and the energy consumption of each *sensor node state* is calculated as the sum of the consumption of the three states that constitutes it. In the same way, the consumption by transitions between *sensor node states* is calculated as the sum of the consumption of the internal transitions involved. To better explain the state transitions consumption, Figure 15 shows an example of a *sensor node state* transition. In this example, the consumption by the transition from *sensor node state 1* to *sensor node state 2*, is equal to the sum of the consumption of the next transitions: CPU_{idle} to

CPU_{active} and Com_{TX} to Com_{sleep}. For this specific example, the sensing domain does not contribute in the energy consumption transition because it remains in sleep mode.

Figure 15: *Sensor node state* transition example



Source: Author

The states possible combinations depend on different factors, for example: the operating system, which may have specific operation or energy aware politics; and the applications programmed in the sensor node, which may implement or may require specific functions. For this reason, to explain the prediction process, we supposed there are *sensor node states*: Sensing, Processing, Transmitting, Receiving, and Low-power mode. The configuration of each *sensor node states* is shown in Table 7, where in the first column are the *sensor node states* and in the first row are the energy consumption domains.

Once the *sensor node states* are defined, the next step is to construct the probability

Table 7: *Sensor node states* configuration

	Sensing	**Processing**	**Communication**
Sensing	active	active	sleeping
Processing	sleeping	active	sleeping
Transmitting	sleeping	active	transmitting
Receiving	sleeping	active	receiving
Low-power mode	sleeping	sleeping	sleeping

Source: Author

matrix. The probability matrix depends on the sensor node's behavior, and also it can vary over time. Before constructing the probability matrix, it is necessary to run a training period. In this period, the sensor node monitors and counts the transitions between all states to create a transitions matrix and a vector with the total transitions of each state. This matrix shows how many times the node remains in the same state after a time-step, and how many times it passes to another state. Equation 5.13 shows the transitions matrix A for five *sensor node states*, enumerated from 1 to 5, where: a_{11} is the number of transitions the node went from state 1 to state 1, a_{12} is the number of transitions the node went from state 1 to state 2 and so on. Equation 5.14 and Equation 5.15 show the total transitions vector V and how to calculate it.

$$A = \begin{pmatrix} a_{11} & a_{12} & a_{13} & a_{14} & a_{15} \\ a_{21} & a_{22} & a_{23} & a_{24} & a_{25} \\ a_{31} & a_{32} & a_{33} & a_{34} & a_{35} \\ a_{41} & a_{42} & a_{43} & a_{44} & a_{45} \\ a_{51} & a_{52} & a_{53} & a_{54} & a_{55} \end{pmatrix} \tag{5.13}$$

$$V = \begin{pmatrix} v_1 & v_2 & v_3 & v_4 & v_5 \end{pmatrix} \tag{5.14}$$

$$\begin{aligned} v_1 &= a_{11} + a_{12} + a_{13} + a_{14} + a_{15} \\ v_2 &= a_{21} + a_{22} + a_{23} + a_{24} + a_{25} \\ v_3 &= a_{31} + a_{32} + a_{33} + a_{34} + a_{35} \\ v_4 &= a_{41} + a_{42} + a_{43} + a_{44} + a_{45} \\ v_5 &= a_{51} + a_{52} + a_{53} + a_{54} + a_{55} \end{aligned} \tag{5.15}$$

Then, the probability matrix P is calculated using the transitions matrix and the

total transitions vector, as shown in Equation 5.16.

$$P = \begin{pmatrix} a_{11}/v_1 & a_{12}/v_1 & a_{13}/v_1 & a_{14}/v_1 & a_{15}/v_1 \\ a_{21}/v_2 & a_{22}/v_2 & a_{23}/v_2 & a_{24}/v_2 & a_{25}/v_2 \\ a_{31}/v_3 & a_{32}/v_3 & a_{33}/v_3 & a_{34}/v_3 & a_{35}/v_3 \\ a_{41}/v_4 & a_{42}/v_4 & a_{43}/v_4 & a_{44}/v_4 & a_{45}/v_4 \\ a_{51}/v_5 & a_{52}/v_5 & a_{53}/v_5 & a_{54}/v_5 & a_{55}/v_5 \end{pmatrix} \tag{5.16}$$

The energy consumption prediction using Markov chain approach will be separated in two terms: E_R is the energy consumption prediction for the usage time in all the energy domains, and E_B is the energy consumption prediction for the transitions. In this way, the energy consumption prediction E_{MC} is shown in Equation 5.17.

$$E_{MC} = E_R + E_B \tag{5.17}$$

To calculate E_R, there are calculated the number of visits to each *sensor node state* using Equation 5.18, explained in Section 2.2.1. As explained in Section 5.1.4, the processing module has a role of master, thus, it is defined an initial distribution $x_0 = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \end{pmatrix}$. Then, the number of visits to each state is multiplied by its average energy consumption. In this manner, the energy consumption E_R for the five *sensor node states* shown in Table 7 can be calculated as shown in Equation 5.18. The term E_s is a vector of the average energy consumption of the *sensor node states*, and T is the prediction time in time-steps.

$$E_R = x_0 \left(\sum_{t=1}^{T} P^t \right) E_s \tag{5.18}$$

To calculate E_B, a transition cost matrix is defined. This matrix is similar to the probability matrix, but showing the energy cost of passing from one *sensor node state* to another. A transition cost matrix B for five states is shown in Equation 5.19, where e_{12} is the energy consumption of going from *sensor node states* 1 to *sensor node states* 2, e_{13} is the energy consumption of going from *sensor node states* 1 to *sensor node* *states* 3 and so on.

$$B = \begin{pmatrix} e_{11} & e_{12} & e_{13} & e_{14} & e_{15} \\ e_{21} & e_{22} & e_{23} & e_{24} & e_{25} \\ e_{31} & e_{32} & e_{33} & e_{34} & e_{35} \\ e_{41} & e_{42} & a_{43} & e_{44} & e_{45} \\ e_{51} & e_{52} & e_{53} & e_{54} & e_{55} \end{pmatrix} \tag{5.19}$$

Then, the number of visits to each *sensor node state* are calculated and multiplied by the vector K_s, which represents the transitions cost. The vector K_s is the result of the element-wise product of the probability matrix and the transition cost matrix, x_0 is the initial distribution vector, and e is a column vector with all entries 1. Thus, E_B can be calculated as shown in Equation 5.21.

$$K_s = P \circ B \tag{5.20}$$

$$E_B = x_0 \left(\sum_{t=1}^{T} P^t \right) K_s e \tag{5.21}$$