Q-Probabilistic Routing in Wireless Sensor Networks

Rocío Arroyo-Valles, Rocío Alaiz-Rodríguez, Jesús Cid-Sueiro, Alicia Guerrero-Curieses

Abstract

Unpredictable topology changes, energy constraints and link unreliability make the information transmission a challenging problem in wireless sensor networks (WSN). Taking some ideas from machine learning methods, we propose a novel geographic routing algorithm for WSN, named Q-Probabilistic Routing (Q-PR), that makes intelligent routing decisions from the delayed reward of previous actions and the local interaction among neighbor nodes, by using reinforcement learning and a Bayesian decision model. Moreover, by considering the message importance embedded in the message itself, routing decisions can be adapted to traffic importance. Experimental results show that Q-PR becomes a routing policy that, as a function of the message importance, achieves a trade-off among the expected number of retransmissions (ETX), the successful delivery rate and the network lifetime.

1. INTRODUCTION AND CHALLENGES

Connecting a source node with a sink node through a route of lossy links in a WSN requires the cooperation among many intermediate nodes. Frequent and unpredictable topology changes as well as the severe energy constraints in sensors, make classical routing techniques fail in these environments.

A wide variety of routing algorithms for ad-hoc and sensor networks have been recently proposed [1]. Connection-based algorithms (both reactive and proactive) differentiate an initial route discovery phase to keep routing tables updated, after which packet forwarding takes place. Consequently, topology changes during the packet transmission phase force to stop forwarding and to start a new route discovery phase. Their performance strongly depends on the topology change frequency what makes them suitable only for very stable networks. Improvements for dynamic networks have been accomplished by “connectionless” [16] methods that carry out route discovery and packet forwarding as a whole, deciding the route at every hop. Search-based, constrained flooding and geographic routing strategies belong to this latter group.

The performance of search-based and geographic routing methods can be seriously affected by physical conditions: the delivery probability between nodes depends on factors such as the distance between them, interferences and collisions, and sending a message between two neighbor nodes may require more than one transmission [13], [4]. The increased robustness provided by flooding-based protocols (where each node individually makes the decision to forward or not) implies, however, an undesirable increment in energy consumption and in the chances of message collision.

A recent opportunistic approach in the context of geographic routing protocols has been proposed [4], [11] in order to manage channel conditions and topology changes. Basically, this approach relies on the fact that the actual forwarding node is chosen after the message has been transmitted among the candidates that successfully received it. It turns out, however, that the coordination between candidate nodes is not straightforward and should be carefully specified. Transmission priorities among candidates are established based on physical distance to destination regardless of the present network state [11] or based on the expected number of retransmissions (ETX) to destination [4], a knowledge that is estimated and distributed by a server at the network startup. Nonetheless, if network operation conditions change, this knowledge should be updated dynamically. Moreover, in a WSN, cost metrics such as node energy availability is also of major concern.

In this paper, we exploit the potential benefits of machine learning techniques to provide sensor nodes with the capability to learn from previous experiences, make routing decisions and adjust to future conditions. Learning reduces the amount of information stored, transmitted and updated, optimize the use of the local information available at nodes and provide a way to deal with uncertainty in the network state.

Our proposal stands on the idea that the cost metrics mentioned above can be learned and estimated from previous traffic patterns and the routing decisions can be updated accordingly. We explore an opportunistic-based routing policy that attempts to learn (i) from the delayed reward (or cost) of previous routing decisions and (ii) from information broadcasted by nodes to their respective neighbors. We will refer to it as Q-Probabilistic Routing (Q-PR).

The paper is organized as follows: Sect. 2 explains the
basic idea underlying the Q-PR algorithm, Sect. 3 provides the details, Sect. 4 contains the experiments and Sect. 5 states some conclusions with some future research lines.

2. **Q-PROBABILISTIC ROUTING: A BIG PICTURE**

The Q-PR algorithm takes advantage of the broadcast nature of transmissions in WSN. The transmitter node sends a message to a group of potential forwarding nodes, though it is likely to be received only by some of them due to lossy links. The selection of the forwarding node results from the coordinated but individual decisions of the nodes in this subset.

Unlike conventional opportunistic routing, (i) the group of candidate nodes is enforced to be neighbors among themselves, what makes coordination easier and with reduced overhead, (ii) the cost metrics are neither a mere distance [11] nor an estimated ETX delivered by a server in the network tuning phase [4], but they are updated from the delayed reward of previously taken actions using a reinforcement learning (RL) rule, and (iii) the message transmission is conditioned to the individual node decisions.

The “Q” term stems from the Q-learning strategy in RL [6]. It has been formerly adapted to the routing problem in wired networks [8], but learning is only carried out on a one-to-one basis: when a forwarding node is selected, it replies to the previous node with its estimated cost metric to destination. Some recent applications of RL to routing in wireless networks [17], [5] follow a search-based approach and therefore, suffer from the same shortcomings. Unlike wired networks, a great advantage can be taken of the intrinsic broadcast nature of a wireless environment that favors the information spread and allows to easily disseminate a more up-to-date network state.

Finally, despite the routing objective in most routing protocols is fixed and independent on the message, embedding objectives into the message allows to apply general-purpose routing strategies that obtain QoS-aware protocols [16]. The Q-PR algorithm uses the importance value embedded in the message itself, to make node decisions about the message transmission. Next, the algorithm is described in depth.

3. **THE Q-PROBABILISTIC ROUTING ALGORITHM**

The Q-PR algorithm discovers the next hop of the actual route (a) as packets travel along the network (and not as a previous phase), (b) after the message has been sent (opportunistic approach), (c) using the geographic position (location-based routing [12]) and (d) making intelligent decisions at the nodes receiving the forwarding requests, based on local information as well as on the importance of message to be forwarded.

### A. Wireless Sensor Network

Consider a static WSN with \( N \) regular source nodes \( N_s, \{i = 1, \ldots, N\} \) and a destination node \( N_0 \) (sink node or access point). We assume that each node is aware of its position, the geographic position of the sink node and local information (state and position) of its neighbors. For details about how nodes can be provided with this information, see [10].

### B. Node Information

As mentioned above, each node \( N_i \) knows its position, \( z_i \), that of the access point, \( z_0 \), and that of its neighbors. Each node also keeps an estimate of the delivery probability with neighbor nodes. Following [14], we assume that the reception probability, \( \alpha(x) \) as a function of distance, satisfies a log-normal shadowing model, given by

\[
\alpha(x) = \begin{cases} 
1 - \frac{1}{2} \left( \frac{x}{R} \right)^2 & \text{if } 0 \leq x < R \\
\frac{1}{2} \left( 2 - \frac{x}{R} \right)^2 & \text{if } R \leq x < 2R \\
0 & \text{otherwise}
\end{cases}
\]

where \( \beta \) is the attenuation factor, and \( R \) is the distance such that \( \alpha(R) = 0.5 \). Delivery probability \( \alpha(x) \) can be also approximated based on signal strengths or statistics from the number of messages sent and received between nodes.

As an example, consider the WSN in Fig. 1. Table 1 shows the local information gathered by node \( N_1 \), and Table 2 its second order neighbors. Node \( N_1 \) is expected to know the following about each neighbor, \( N_j \):

- \( D_{ij} = \|z_j - z_0\| \): Euclidean distance to destination.
- \( \hat{P}_{ij} = \hat{P}_{ij} \), where \( D_{ij} \) is the distance between nodes \( N_i \) and \( N_j \): Delivery probability between nodes.
- \( \hat{Q}_{ij} \): Estimation of the ETX to reach destination node \( N_0 \) from \( N_i \) through neighbor \( N_j \).
- \( \hat{E}_{ij} \): Estimation (by \( N_i \), based on the number of overheard retransmissions) of the energy at node \( N_j \).

![Table 1: Local Information of \( N_i \)'s Neighbor Nodes.](image)

<table>
<thead>
<tr>
<th>Local information at ( N_i )</th>
<th>Neighbor nodes ( N_j )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( N_2 )</td>
<td>( N_3 )</td>
</tr>
<tr>
<td>Distance to sink, ( D_{ij} )</td>
<td>7</td>
</tr>
<tr>
<td>Delivery prob., ( \hat{P}_{ij} )</td>
<td>0.55</td>
</tr>
<tr>
<td>ETX to sink, ( \hat{Q}_{ij} )</td>
<td>2.5</td>
</tr>
<tr>
<td>Estimated Energy, ( \hat{E}_{ij} )</td>
<td>0.7</td>
</tr>
</tbody>
</table>

The neighbors of node \( N_i \) will be denoted by \( \phi_i \), and those closer to the access point than the node itself by \( \phi_i^c \).

Unlike algorithms that consider neighbors to all nodes within a distance \( R \) (UDG, Unit Disc Graph model) regardless of the link quality, our algorithm is aware of the delivery probability within nodes. As pointed out in [13], the UDG can be improved and replaced by a more realistic model, where packet reception probability depends not only on the distance between nodes but also on other factors, such as interferences. Thus, we consider two nodes as neighbors if the (estimated) delivery probability between them is above a predefined threshold.

### C. Making a routing decision

1. **Selecting forwarding candidates:** Q-PR is a greedy position-based algorithm, based on the cost metric given by

\[ Q_{ij} \]

Note that the cost metric, commonly referred as Q-value in RL, is denoted here by \( Q_{ij} \) and it refers to ETX.
$CM = Q/P$, where $Q$ is the expected number of retransmissions (ETX) to destination and $P$ is the progress (by how much the distance to destination would be decreased) in that hop. Taking into account that ETX integrates energy consumption, link quality and delays, optimizing ETX [3] seems more suitable than minimizing the number of hops.

When node $N_i$ receives a message, it is notified to be responsible (together with other neighbors of higher and/or lower priority) for forwarding it to the sink node. If higher priority nodes forward the message, node $N_i$ will not do it. Otherwise, the node has to make a discard/transmit decision.

To do so, it proceeds as follows: (i) from the set of nodes closer to destination than itself, it selects the node minimizing the cost metric $Q/P$, and (ii) iteratively selects the best nodes among the remaining ones that fulfill the constrain of being neighbors of the previously selected candidates. This latter point assures that the set of candidate forwarding nodes can communicate among themselves. This subset of candidates selected by node $N_i$ will be denoted as $\phi_i^*$. 

In the scenario previously described, with $N_1$ at 9 unit distance to destination ($D_{10}=9$), the subset $\phi_1^* = \{N_2, N_3, N_7, N_8\}$. According to the cost metric in Table 3, the ordered set becomes $\{N_7, N_2, N_8, N_3\}$. However, according to Table 2, node $N_8$ is discarded since it is not neighbor of both previously selected candidate nodes, what results in $\phi_8^* = \{N_7, N_2, N_3\}$ with decreased order of optimality and therefore, priority to forward the message.

In order to get the compromise between route exploitation-exploration and following a common approach in RL methods [6], the actual forwarding node is selected at random among all neighbors, with a small probability $c$.

A special situation arises when the candidate set is empty (i.e. there are no progressing neighbors). In this case, a list of candidates, prioritized by the delivery probability, is created, placing the node from which the message is received at the end. The same applies when neighbor nodes become non-cooperative, but in this case selection as forwarder node the following in the original prioritized candidate list.

2) Transmit/discard decision: After selecting the set of candidate neighbor nodes, $\phi_i$, a decision has to be made about forwarding the message to these nodes, taking into account the energy expense and the probabilities that they will eventually forward the message. Our decision model is a modified version of a preliminary model in [2], and is explained below:

Node $i$ decisions will be based on:

- Single Transmission Energy: $E_T$
- Single Reception (or idle mode) Energy: $E_{IR}$ (we assume equal energy expenses in reception and idle modes [15]).
- Unlike [2], here we assume more energy consumption in transmission than in reception (or idle) mode.
- Message Importance: $I$
- Candidate profiles, learnt from previous experiences

All these variables are grouped into an observation vector $x$ and, based on it, the node decides whether to forward the message ($d = 1$) or not ($d = 0$), following a Bayesian decision approach with two hypothesis: $T = 0$ and $T = 1$, where:

$T = 0$, if no candidate node will forward the message (thus, saving energy to transmit more important messages).

$T = 1$, if at least one of the candidate nodes will forward the message.

If $c_{ij}$ is the cost of deciding $d = i$ when $T = j$ (where $i, j \in \{0, 1\}$) we define: $c_{00} = E_{IR}, c_{01} = E_{IR}, c_{10} = E_T$ and $c_{11} = E_T - \gamma I$. Note that the cost of rejecting a forwarding request is the energy spent while keeping listening, and $c_{11}$, the cost of deciding to forward a message when there are candidates to retransmit, is reduced according to message importance $I$. Parameter $\gamma$ modulates the trade-off between the transmission energy (cost) and the message importance (benefit), unlike the model in [2]. Nevertheless, the cost model proposed here has an illustrative purpose.

According to this, the risks (mean costs) associated to decide in favor to transmit or discard are given by

\[
R(d=1|x) = E_T - \gamma I P(T=1|x) \tag{2}
\]

\[
R(d=0|x) = E_{IR} \tag{3}
\] 

respectively, and the minimum risk decision rule [20] is given.

### Table 3: Cost metric of $N_i$'s Neighbor Nodes.

<table>
<thead>
<tr>
<th>Neighbor nodes</th>
<th>$N_1$</th>
<th>$N_2$</th>
<th>$N_3$</th>
<th>$N_7$</th>
<th>$N_8$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Distance to destination: $</td>
<td>x_i - x_0</td>
<td>_1$</td>
<td>10</td>
<td>8</td>
<td>3</td>
</tr>
<tr>
<td>ETX to Destination: $C[Q/P]$</td>
<td>23</td>
<td>2</td>
<td>8</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>ETX/Progress: $</td>
<td>x_i - x_0</td>
<td>_1 -</td>
<td>x_j - x_0</td>
<td>_1$</td>
<td>1.25</td>
</tr>
</tbody>
</table>

### Table 2: 2-hop Neighbors: Neighbors of $N_1$'s Neighbor Nodes.

<table>
<thead>
<tr>
<th>Neighbor nodes</th>
<th>$N_2$</th>
<th>$N_3$</th>
<th>$N_4$</th>
<th>$N_5$</th>
<th>$N_6$</th>
<th>$N_7$</th>
<th>$N_8$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$N_2$</td>
<td>$N_3$</td>
<td>$N_4$</td>
<td>$N_5$</td>
<td>$N_2$</td>
<td>$N_1$</td>
<td>$N_4$</td>
<td>$N_7$</td>
</tr>
<tr>
<td>$N_3$</td>
<td>$N_4$</td>
<td>$N_2$</td>
<td>$N_5$</td>
<td>$N_3$</td>
<td>$N_1$</td>
<td>$N_7$</td>
<td>$N_8$</td>
</tr>
<tr>
<td>$N_4$</td>
<td>$N_2$</td>
<td>$N_3$</td>
<td>$N_5$</td>
<td>$N_3$</td>
<td>$N_1$</td>
<td>$N_7$</td>
<td>$N_8$</td>
</tr>
<tr>
<td>$N_5$</td>
<td>$N_3$</td>
<td>$N_2$</td>
<td>$N_5$</td>
<td>$N_1$</td>
<td>$N_7$</td>
<td>$N_8$</td>
<td>$N_4$</td>
</tr>
<tr>
<td>$N_7$</td>
<td>$N_4$</td>
<td>$N_5$</td>
<td>$N_3$</td>
<td>$N_1$</td>
<td>$N_8$</td>
<td>$N_4$</td>
<td>$N_8$</td>
</tr>
<tr>
<td>$N_8$</td>
<td>$N_1$</td>
<td>$N_5$</td>
<td>$N_3$</td>
<td>$N_4$</td>
<td>$N_8$</td>
<td>$N_4$</td>
<td>$N_8$</td>
</tr>
</tbody>
</table>
The probability that candidate node $N_i$ transmits a message depends on the decisions made by other candidate nodes with higher priority in the list. In order to decide whether to transmit or not, node $N_i$ estimates the posterior probability of hypothesis $T = 1$. Defining $P(F_j = 1|x)$ as the posterior probability that $N_j$ is able to forward the message and $P(T_j = 1|x)$ as the posterior probability that $N_j$ will definitely retransmit the message, we have

$$P(T = 1|x) = \frac{1}{\gamma I} \sum_{k=0}^{\infty} \frac{E_T - E_{1R}}{E_T - E_{1R}}$$

(4)

The information captured from the environment or from other nodes is used to update the ETX to reach $j$ from $i$. Though a decreasing $\eta$ is usual in RL algorithms, we keep it constant for a fast adaptation to the dynamic network behavior.

The ETX from $i$ to $j$ can be estimated as

$$\hat{q} = \sum_{k=1}^{\infty} k(1 - \hat{P}_{D_{ij}})^{(k-1)} \hat{P}_{D_{ij}} = \frac{1}{\hat{P}_{D_{ij}}}$$

(10)

Note that $Q_j(x)$ depends on $j$, which is the eventual forwarding node. The ETX (over all candidate nodes) to sink node must be computed as a mean over all possible forwarders,

$$\hat{q}_i(\phi^*_i) = \frac{\sum_{N_j \in \phi^*_i} P(T_j = 1|x)Q_j(j)}{\sum_{N_j \in \phi^*_i} P(T_j = 1|x)}$$

(11)

Unlike RL algorithms that update the cost metric at node $N_i$ with the minimum cost to destination, here we update it with an average cost since next forwarder is not known in advance.

2) Learning neighbor profile.

Each node estimates parameters $w$ of the logistic neighbor profile in (8) without exchanging any information among nodes, but just by overhearing node retransmissions.

Defining $y_{ij} = P(F_j = 1|x)$ and $d_{ij}$ as the decision of node $N_j$ to forward a message from node $N_i$, parameters $w_j$ are estimated in order to minimize a loss function able to provide adequate estimates of the posterior probabilities to forward the message, such as the cross entropy loss function [9], commonly used in neural network training algorithms,

$$L(y_{ij}, d_{ij}) = -d_{ij} \ln y_{ij} - (1 - d_{ij}) \ln(1 - y_{ij})$$

(12)

Applying stochastic gradient learning rule, node $N_i$ computes

$$w_j^{(k+1)} = w_j^{(k)} + \mu(d_{ij}^{(k)} - y_{ij}^{(k)}) \frac{\hat{P}_{D_{ij}} - y_{ij}^{(k)}}{\hat{P}_{D_{ij}}(1 - y_{ij}^{(k)})} \hat{q}_i$$

(13)

E. Updating information from statistics

Assuming that energy consumption in sensor networks is similar in reception and idle modes [15] and lower than energy consumed by transmissions, $\bar{E}_{ij}$, the energy at node $N_j$ can be estimated by node $N_i$ (whenever it hears a retransmission by node $N_j$ or when it is aware $N_j$ receives a message) as

$$\bar{E}_{ij}^{(k+1)} = \bar{E}_{ij}^{(k)} - d_{ij}^{(k)} E_T - (1 - d_{ij}^{(k)}) E_{1R}$$

(14)

There are, however, messages received by $N_j$ that node $N_i$ is not aware of. Experimental results show that this slight energy overestimation seems to affect neighbor nodes concentrated in a local region in a similar way, so it results not significant to make routing decisions.

F. Q-PR Algorithm

The Q-PR algorithm is summarized as follows,

FORWARDING PHASE ($N_i$ receives a message with a list of other candidates)

if Candidate nodes with higher priority forward the message then

$N_i$ discards the message

else

1. $N_i$ selects the potential forwarding subset $\phi^*_i$
2. \( N_i \) evaluates \( P(T = 1|x) \) according to (5) and makes decision \( d_i \) whether to transmit or not according to (4)

\[
\text{if } d_i = 1 \text{ then}
\]

- Broadcast the message destined to \( \phi_i^* \)
- Broadcast its expected \( \hat{Q}_i(\phi_i^*) \) computed as (11)
- Listen to retransmissions and update candidate neighbor profiles according to (13)

\[\text{if } \text{no candidate nodes forward then} \]

- go to step 1.

\[\text{end if} \]

\[\text{end if} \]

INFORMATION UPDATING PHASE

\( N_i \) updates \( \hat{Q}_i(j) \) via RL according to (9) from any neighbor \( N_j \) that broadcasts its cost metric.

\( N_i \) updates \( E_{ij} \) according to (14) when it hears a retransmission by \( N_j \) or it knows \( N_j \) receives a message

4. Simulation results

Experimental results are carried out over a network topology with 100 homogeneous nodes with equal initial energy resources, that have been randomly deployed in a test 10 \( \times \) 10 area. Node energy is initialized uniformly and power consumption ratios of idle:receiving:transmission are set to 1:1:2.5.

Parameter \( \gamma \) in Eq. (4) was fixed to 1. We assume as neighbor nodes those with delivery probability \( \alpha > 0.05 \).

Note that loss links are also considered in these simulations according to the model presented in Sect. 3-B with \( \beta = 2 \) and \( R = 2.5 \) (a perfect channel knowledge is assumed). Parameters such as \( \mu \) (adaptive step of the weight vector in (13)), \( \epsilon \) (explained in Sect. 3-C.1) and \( \eta \) (learning rate to update the Q metric in (9)) are fixed to 0.05, 0.05 and 0.01, respectively, and remain constant with time. We consider unicast transmissions from randomly chosen sensors to the sink node, represented by the rightmost node in the field. Sensors transmit messages with an importance value selected at random as an integer between \( I = 1 \) and \( I = 10 \). Our Q-PR algorithm is compared to GPSR (Greedy Perimeter Stateless Routing) [18][7], a well-known and widely analyzed position-aware routing algorithm in sensor networks and to EFE (Expected progress-Face-Expected progress) [19], which considers a realistic physical layer. Results were averaged over 50 different topologies of connected networks.

A major problem in routing algorithms based on greedy forwarding occurs when the forwarding node has no neighbors closer to the destination than itself. GPSR tries to solve it by means of perimeter forwarding on the planarized network graph using the right-hand rule, just like EFE does. RNG (Relative Neighborhood Graph) planarization has been implemented [7], Q-PR tackles it by means of the aforementioned prioritized list of candidates as a function of the delivery probability. Fig. 2 illustrates this: a route has to be found from \( N_{14} \) to the sink \( (N_{54}) \). When the message reaches node \( N_{28} \) (following the thick line), no neighbors are closer to the sink than itself. However, the message successfully reaches again one of the ways to destination, coming back to node \( N_{78} \).

In order to simulate routing protocols under the same circumstances, we have made GPSR more robust establishing a maximum number (equal to 20) of transmission retries before discarding the message, as considered in [7] for cases of mobile networks. In EFE, the sender node keeps transmitting the packet (up to 20 retries) to the subsequent candidate until the packet is acknowledged successfully. Q-PR performance is assessed in terms of ETX and the metrics shown in Table 4, where network lifetime is defined by the time slot when there is no route to reach the sink node, and the Received Importance Ratio (RIR) is the ratio between the sum of received importances and the total importance generated by the network traffic.

Results recorded in Table 4 show that Q-PR outperforms GPSR in terms of successful delivery rate (98.12\% against 75.66\%) but it is similar to EFE (98.04\%). Moreover, due to the high number of retransmission retries and acknowledgements (6.08 for Q-PR, 15.82 for GPSR and 10.50 for EFE on average) as well as the high message loss in the case of GPSR, the Q-PR algorithm allows the nodes to keep energy for longer time. Thus, the network lifetime with Q-PR is 751 slot times vs. 203 for GPSR and 174 for EFE. An example of the temporal evolution of ETX to destination is shown in Fig.3. Note that GPSR and EFE allow to transmit to the sink much less than half (26.37\% for GPSR and 29.28\% for EFE, on average) of the messages Q-PR does (154 and 171 against 584). Note also that, although GPSR allows to reach
5. Conclusions and further directions

Q-Probabilistic Routing (Q-PR) is an energy-aware routing algorithm for wireless sensor networks which incorporates intelligent routing decisions adapted to the network dynamics by: (1) estimating a global cost metric locally learned through RL rules during neighbor interactions, and (2) taking forwarding decisions by means of a Bayesian decision model.

Experimental results show that Q-PR allows to increase successful message transmissions as well as to keep a trade off between network lifetime and ETX. Moreover, transmission of least important messages is suspended in favor of high priority messages by considering the message importance in routing decisions, what is seldom studied in the literature.

The result analysis reveals that Q-PR could be further improved by means of a dynamical adjustment of parameter $\gamma$ in Eq. (4) (balancing the message importance with the energy cost), and a refinement in the residual energy estimation at neighbor nodes. Both of them as well as exploring alternative methods to select forwarding nodes that do not require neighborhood requisites are some of the goals for future work.

Acknowledgments

This paper has been partially supported by Spanish MEC projects TEC2005-06766-C03-01/TCM and -2/TCM, CAM-UC3M project S-MORSE CCG06-UC3M/TIC-0805, and CAM project P-TIC-000223-0505.

References