Quality of Information: an Empirical Approach

Erol Gelenbe and Laurence Hey
Intelligent Systems and Networks Group
Department of Electrical and Electronic Engineering
Imperial College London, London SW7 2BT, UK
{e.gelenbe,laurence.hey}@imperial.ac.uk

Abstract

In this paper we examine the Quality of Information (QoI) at the output of a wireless sensor network by considering the difference between the monitored environment and the interpreted data produced by the network. Using practical examples in an experimental setting, we hope to shed light on the concept of QoI and on the manner of estimating and evaluating it. We use a real wireless network in combination with simulated events, to help us formulate and understand the concept of QoI and its associated technical questions. Using algorithms such as trilateration and clustering to interpret the outputs of the sensor network, we explore several definitions of QoI, including the peak signal to noise ratio. Furthermore we investigate the impact that different packet transmission approaches have on the QoI. We show that QoI is time-varying, and that in-network processing allows QoI levels to be maintained while reducing network load.

1. Introduction

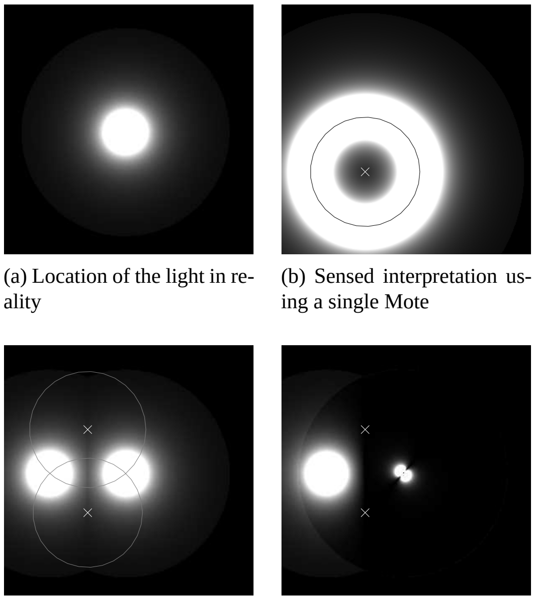
The Quality of Information (QoI) of a sensor network (SN) [1] can be viewed informally as the difference between the data that the output of the SN produces concerning some "environment" that is being monitored, and the actual events in that environment which one wishes to observe or track. Although various formal definitions about QoI have been proposed [2, 3, 6], it is fair to say that there is no agreed overall definition that is as well understood or as well accepted as, say, the definitions of Quality of Service (QoS) in communication networks, or Quality of Images in computer vision and image processing. Indeed, one may consider that QoI results from a combination of the positioning and capabilities of the sensors being used, the effect of the network itself in terms of data loss, delay and so on, and also the algorithms that may be used to fuse the data coming from sensors which are being simultaneously used to observe the environment.

In view of these challenges, in this paper we have chosen to consider simple practical examples in an experimental setting, in the hope that our pragmatic approach may shed light on the concept of QoI and on the manner of estimating and evaluating it. Although the approach we describe is based on our own work within a laboratory environment, the fact that we use a real wireless network (with some thirty-four Motes), in combination with simulated events, should help us better understand the concept of QoI and the interesting technical questions that it raises.

2 The Experimental Setting

The setting that is being considered includes a number of wireless sensing Motes, each of which contains a light detector. The radio of the Motes allows each Mote to send packets to its neighbours, and over multiple hops all the way to a single output "sink" node which collects all the packets that it receives [5]. We switch small lights of fixed intensity on and off in the area of the SN, and the intensity of each light follows an inverse square law over the distance from their location. When a light is on, the Motes which are close enough to the light can sense that it is on, and approximately measure the intensity of the light, but cannot sense the direction from which the light is coming. The sink Mote can in principle deduce the range of a particular light to the Mote that is reporting it, provided that only one light turns on at a time.

In Figure 1(a) we show the case where one light is on. We also show how a single reading causes this to be interpreted at the receiver in Figure 1(b): all that the output sink can tell is that there is a light that is on and that it is located on some approximately defined circle with the Mote at its centre. With readings from two Motes, the information at the output of the SN is in the form of the **intersection** of two approximately defined circles, as shown in Figure 1(c). The approximation is due to the error in our knowledge of the intensity setting of the lights themselves, and also due to



(a) Location of the light in reality

(b) Sensed interpretation using a single Mote

(c) Sensed interpretation using a two Motes

(d) Difference between (a) and (c)

Figure 1. Example interpretation at the sensor network output of one or two readings. The location of the Motes reporting measurements is shown by small crosses.

the errors in the measurements made by the light sensors. It is easy to imagine how the information at the output of the SN becomes difficult to interpret if there are many Motes, and many light sources which may be on simultaneously. Note that Figure 1(d) is a crude QoI metric: just the absolute value of the error between images in Figure 1(a) and (c), where more light indicates higher error.

Note that in this paper we use "lights" as a surrogate source of radiation instead of more sophisticated means such as other wireless radio signals.

2.1 Defining QoI in this Simple Context

Consider an external observer who has a birds eye view of the environment we have just described. Let us suppose that this observer is able to capture a video sequence from a fixed downward looking camera on the ceiling of the room. The potentially blinking lights are placed on the floor, and the observer's video sequence will be the **ground truth** G_t. Thus, thanks to this video, and assuming that the frame rate is high enough, at any time t we know exactly what happens, and we have a record of the state of all the points on the usually dark floor.

Now suppose that each Mote reports back to the sink a light intensity level that it has measured. Suppose that at the sink there is an algorithm which tries to recreate the image that is reported by the camera, but of course with the imperfections that we have outlined: any point of light sensed by a single Mote will be reported as a circle, and so on. Furthermore all the data arrives at the sink after some random delay. Even assuming that (1) only one light is on at a time, (2) none of this data gets lost, and (3) that the sink can process each arriving data packet in a known fixed time to update the ongoing video that the sink is creating as its "fused" output, we will have a SN output that is a video sequence V_t which will be a "pale approximation" of G_t.

Now consider the two frames G_t and V_t at the same time instant t; we can think of G_t as the signal, while V_t is the "signal plus the noise". Thus we define the **mean square error** or noise M_t, and the **peak signal to noise ratio** (PSNR) Q_t, at time t as:

$$M_t = \frac{\sum_{i=1}^{I}\sum_{j=1}^{J}[G_t(i,j) - V_t(i,j)]^2}{I \times J} \tag{1}$$

$$Q_t = \frac{max\{G_t(i,j) : 1 \leq i \leq I, 1 \leq j \leq J\}}{M_t} \tag{2}$$

where $G_t(i,j)$ and $V_t(i,j)$ are the pixel values of G_t and V_t respectively, and $I \times J$ is the size of the frame. Note that while the PSNR is often expressed in decibels, we have chosen to use a pure ratio for our definition.

We may use either Q_t or M_t as the QoI for the information produced by the SN at time t. However this is admittedly a crude definition for a number of reasons. First of all, it considers errors all over each of the video frames, while what we may really care about is the accuracy of the location of the point where the lights are turning on. This can be easily taken care of by thresholding the content of each frame G_t and V_t so that only bright areas and the differences in the location of the bright spots are kept. Secondly, one may wish to be more sophisticated, and consider both errors (or accuracy) in detection and false alarms. Clearly this discussion has not exhausted, by far, the question of how QoI can be defined but it hopefully provides some insight into the factors we may consider, and some of the quantitative approaches that can be pursued.

2.2. Some Experiments with QoI

In the approach that we have just suggested we see that in practice QoI will be a time varying quantity, which will also depend on network delays and losses, computational delays, and on the algorithms that are used to obtain the view V_t of the data obtained at the SN output. Obviously V_t would have to somehow integrate time dependent information, and the resulting QoI would also depend on the speed at which the observed situation changes in the "real" environment.

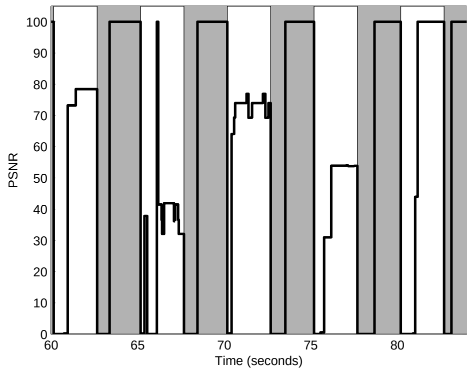


Figure 2. A time sample of the PSNR from the first experiment. The “light” and “dark” areas indicate when the light is on and off.

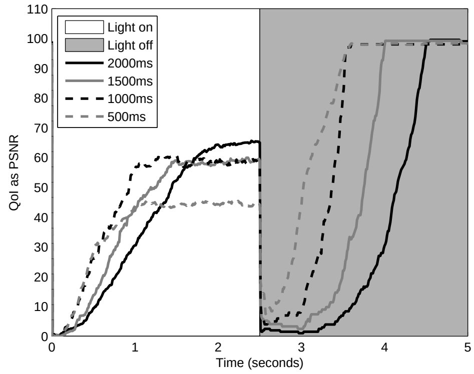


Figure 3. An ensemble average of the PSNR over the light’s on and off cycles. The result for four different sample intervals are shown.

We now propose a number of experiments which, although simplistic in nature, attempt to capture a range of properties and influences that the sensor network has on the outputs.

In the first experiment, the setting includes 20 wireless sensing Motes, placed at regular intervals of $100cm$, in a $4m \times 5m$ rectangular grid. A single light turns on and off *every two and a half seconds* in a random location within the area of the SN. The light has a brightness which can be sensed by a Mote up to a distance of $125cm$, but not beyond. The Motes themselves take a measurement *every one second* and transmit the reading over multiple hops to the sink. Readings arriving at the sink are kept for one second after they are received and are used to create the view V_t. Figure 2 shows a sample of the time varying PSNR resulting from this approach. The “light” and “dark” areas of the graph illustrate when the light is on and off. We can see that there is a delay between the light turning on and the arrival of measurements at the SN output (and the subsequent increase in the PSNR or QoI). Lost or delayed packets also mean that we do not always have three readings to faithfully reproduce the ground truth.

As each Mote is sampling each second and forwarding its readings, light readings of zero are also received at the SN output. These readings do not give us the location of a light, but can be used to determine the absence of light. This is especially useful when only two positive Motes have returned readings indicating the presence of light. As two readings place the light in one of two locations at the intersection of the circles created by the readings, a reading of zero from a third Mote will help to determine which of the two intersections is *not* the location of the light. This increases the QoI.

When the light is off, we treat the absence of light as our signal by inverting the frame in G_t. As readings arrive at the sink with readings of zero, or as existing readings timeout, we can recreate this frame perfectly so the PSNR rises to 100 (the limit we have arbitrarily applied).

When three readings are being used to position the source of the light, we could use the offset between the actual location and the sensed location as a QoI metric. When the output of the SN has given us enough data to place the light in one of two locations we could use the sum of the separation between the real position and the two sensed positions as the measure of QoS. However with a single reading giving us a continuous circle, this method ceases to function. This is where, as a measure of the QoI, the advantages of the PSNR are apparent. In giving us a continuous measure of the QoI at the output of the SN, applicable in many situations, the PSNR is a versatile metric.

From Figure 2 we observe a further measure which can be applied to the QoI. The **lag**, i.e. the difference in times at which the event occurs in the environment and is reflected in the measurements, is a function of the sampling rate of the Motes, the network delay, and the packet loss probability of the links of the network. Figure 3 shows an ensemble average (over one “on” and “off” period) of the PSNR, for four sampling periods. We observe that as the sampling interval decreases, the *lag* between the “event” (i.e. the light turning on or off) and the subsequent rise in the QoI also decreases. However, we also observe that, for a sampling interval of $500ms$, this benefit is not evident, and that additionally the QoI achieved is lower than in the other cases. The extra load this sampling rate places on the network results in a significantly higher packet loss rate of 0.48, compared to 0.07 with a $2000ms$ sampling interval.

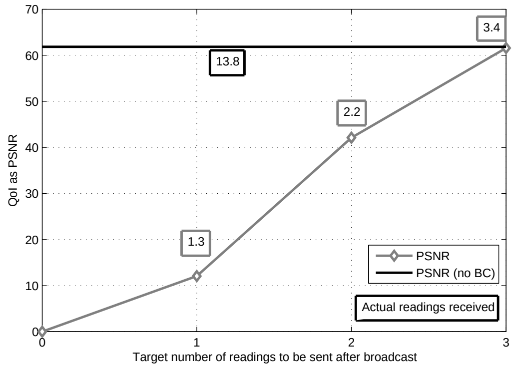


Figure 4. QoI resulting from only the top n readings, possible due to broadcasts. The mean number of readings used for each value of n is shown above the PSNR values.

In order to further illustrate the SN's effect on the interpretation of the readings, we have run experiments using a different approach to packet transmission. In the first experiment, all Motes were sampling the light level at one second intervals and sending their readings to the sink. For this second approach our aim is to reduce the overall network traffic. When a Mote senses the presence of light it broadcasts its reading so that other Motes in the area, which may have also sensed the same light, receive it. This allows the Motes to intelligently determine whether to transmit their readings to the SN output. For example, only the Mote with the strongest reading (logically the Mote closest to the origin of the light) may transmit its reading, or if the computational abilities of the SN allow, individual Motes may attempt to estimate the QoI from the readings in order to intelligently decide whether their readings are required to meet some QoI threshold. These approaches should reduce the number of packets travelling on the multi-hop route to the SN output as well as the power consumption.

In Figure 4 we show results from experiments where we allow the Motes with the top n readings to send their readings. The PSNR values shown are the mean of those achieved by the time just before the light turns off. The figure also shows the mean number of readings received at the SN output at that point in time. The results from the previous experiment are shown for reference. We see that, when using broadcasts, there is an approximately linear relationship between n and the QoI. Also evident is that the number of readings received at the sink is slightly higher than n. This occurs as some Motes may not receive all broadcasts, and therefore transmit their readings despite not having one of the n strongest readings. However, the numbers are still considerably lower than the mean of approximately 14 readings for the case without broadcasts, indicating a substantial power saving.

2.3. Interpretation of PSNR in this Context

One issue yet to be addressed is the interpretation of a particular PSNR value with respect to the QoI. For a single reading the PSNR is a non-linear measure of the distance between the sensor and the origin of the light. The PSNR ranges between close to zero at $125cm$, and 10 at a distance of $5cm$, and as the distance approaches zero the PSNR approaches infinity. When the sensed interpretation places the light in one of two locations, and when the two locations are significantly separated, the PSNR is dominated by the "incorrect" location. Assuming the "correct" location matches perfectly, we achieve a PSNR of just below 60. As the two locations begin to converge this value rises. A third reading places the light at a single point, and the PSNR is a measure of the distance between the real location and the sensed location. In this case we expect a PSNR between 30 (when the distance approaches the sensing range) and 100 (our arbitrary limit). We may therefore heuristically state that, when monitoring a single light, a PSNR above 60 is excellent as it indicates an accurate location of the origin of the light, and a PSNR above 30 is good.

3. Extending the Experimental Setting with Multiple Simultaneous Lights

In Section 2, the experimental setting was limited to a single light or event occurring within the monitored area. This was an aid to the introduction of some of the concepts of QoI used in the previous experiments. A more realistic scenario would extend to the more challenging case where the SN is used to monitor a less constrained environment in which a number of events may be occurring simultaneously. In this section we explore several such scenarios, and in each case propose an algorithm with which to process the SN outputs.

3.1. Lights in Random Locations

Here we incorporate multiple lights which may be on simultaneously. We ensure that the lights are sufficiently far apart, so that each Mote only senses one particular light. We must therefore attempt to group readings from different Motes into sets associated with individual lights before performing the trilaterations. As each Mote reading gives us only data on distance, and not direction, and as readings may originate from lights which are no longer on due to network and sampling delays, this grouping is non-trivial.

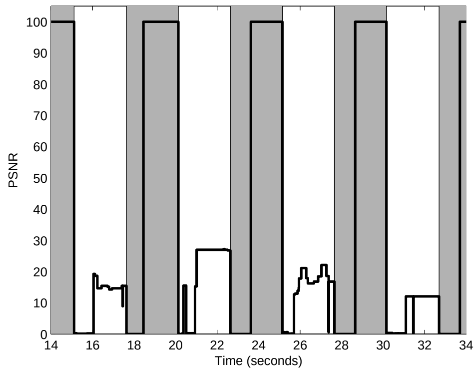


Figure 5. A sample from the experimental results of Section 3.1 with multiple randomly placed simultaneous lights.

In this experiment, as previously, the Motes are sampling the light at one second intervals, and sending their readings to the sink. The algorithm which we employ at the sink has several steps. We initially apply a simple spacial clustering algorithm in order to separate groups of readings which cannot be related due to distance. For each of these spacial clusters, we then generate a "pair-matrix". Recall that a single reading places the origin of the light on a circle centred around the Mote. The pair-matrix determines which Motes' reading-circles intersect, indicating that they *may* be readings of the same light.

We calculate an estimate for the number of lights within each spacial cluster by recursively determining the largest number of "unpaired" readings; in other words, readings which *must* be of different lights. These readings are dubbed "anchor" readings and simplify the next step of the algorithm. For each anchor reading we determine the largest "pair-set". Each reading paired with an anchor reading results in two intersect points. A "pair-set" is a set of readings paired with the anchor which all match a particular intersect point. If the anchor has no paired readings then the pair-set size is one. If there is only one paired reading or each paired reading has no other matching readings then the pair-set has a size of two. Each intersect point is checked for "zero"-readings from nearby Motes which make the point unlikely.

The anchors are sorted by the size of their largest pair-set. The size of the pair-set reinforces the location of the light, and thus a cluster is created with the readings from the largest pair-set. If there is more than one anchor with that size of pair-set, the anchor with the smaller number of pair-sets takes precedence. The Motes are removed from the spacial cluster, and we recalculate the largest pair-sets for each anchor. This is repeated until there are either no more Motes in the spacial cluster, or all the anchors are used.

This algorithm allows us to perform the successive stages of trilateration on each cluster of readings so as to recreate the frame. As previously, to evaluate the use of the PSNR as a QoI metric, we have run experiments on our SN testbed. Figure 5 illustrates a sample of the time-varying PSNR as three lights turn on and off. Other than the number of lights, the setting matches that of the previous experiments. From the figure it is clear that the average PSNR is lower than for the single light example. This is expected as with multiple lights there are multiple locations for error or noise in the recreated frame. This is an issue that arises in quantifying the QoI from multiple events in this manner.

3.2. Lights in Fixed Locations

The setting for this experiment envisions a scenario where there are multiple lights with fixed locations placed over an area being monitored by Motes, whose number is smaller than the number of lights. This scenario may arise if, for example for cost reasons, the number of Motes with wireless radios and processing capabilities is limited, and events are detected using Triggers which communicate with the Motes via lights or other forms of radiation or signalling. By using the sensed light reported by the Motes to calculate which Triggers are On and Off, we can determine where in the monitored area the activity is taking place.

The readings reported by each Mote are the sum of the intensities of the light from each Trigger at that Mote. As both Motes and Triggers have fixed locations, and the light intensity decreases over distance in a predictable manner, the intensity which each Mote should sense from each Trigger which is On is known.

Let r_m be the reading at Mote m, and let $i_{m,\tau}$ be the contribution from Trigger τ to Mote m, assuming τ is On. Let x_τ be a binary value determining if the Trigger τ is On or Off. Let M be the number of Motes, and T the number of Triggers. We then have a system in the form $\mathbf{r} = I\mathbf{x}$, where $\mathbf{r}$ is a vector of the readings r_m, I is the matrix of known light values $i_{m,\tau}$, and $\mathbf{x}$ is the binary vector of the Triggers' states x_τ which we wish to calculate. This is given in (3).

$$\begin{pmatrix} r_1 \\ r_2 \\ \vdots \\ r_M \end{pmatrix} = \begin{pmatrix} i_{1,1} & i_{1,2} & \cdots & i_{1,T} \\ i_{2,1} & i_{2,2} & \cdots & i_{2,T} \\ \vdots & \vdots & \ddots & \vdots \\ i_{M,1} & i_{M,2} & \cdots & i_{M,T} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_T \end{pmatrix} \tag{3}$$

As there are more unknowns than equations in this linear system, it cannot be solved uniquely. Furthermore, as the network impacts the accuracy of the values in $\mathbf{r}$, which may be delayed or lost, there may be no solution which satisfies all the equations at any one point in time. A common

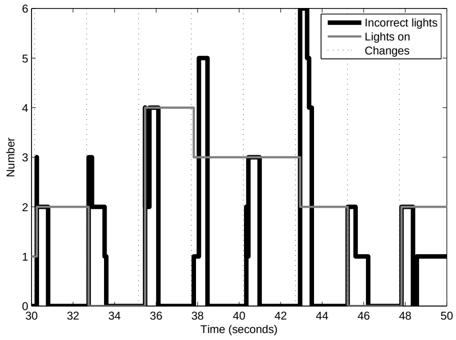


Figure 6. A sample of the results from Section 3.2. Number of incorrectly detected lights and the actual number of lights in the On state are shown.

method for solving such underdetermined systems is the Moore-Penrose pseudoinverse, which is applied to matrix I to calculate a least squares solution [4].

Before using the pseudoinverse to determine which Triggers are On or Off, we attempt to minimise the number of unknowns and thus increase the performance. Let $L(\tau)$ be the set of all (other) Triggers except τ. Then, for each Mote m, if there is a recent reading:

- if $r_m < i_{m,\tau}$ then Trigger τ is obviously Off,
- if $r_m \geq i_{m,\tau}$ and $i_{m,\tau} > \sum_{l \in L(\tau)} i_{m,l}$ then τ is On.

The Triggers with known states are then removed from the system in (3), and the Triggers we have determined to be On have their known light contributions $i_{m,\tau}$ subtracted from the Motes' readings. These adjusted readings are then used in the pseudoinverse calculation to determine the state of the remaining Triggers.

Our example uses six Motes arranged in two columns of three, and twelve Triggers arranged in three columns of four between the Motes. Every 2.5 seconds each Trigger has an independent probability of 0.1 of switching On if Off, and 0.75 of switching Off if On. A sample window from the results is shown in Figure 6. In this example we have used the number of incorrectly determined Triggers as a measure of the QoI. As in the previous examples, the PSNR could be employed, yet would simply be proportional to the number of correctly interpreted Triggers, while obfuscating the information. It therefore makes sense to use a scenario specific metric.

As in the previous experiments, we see that there is a *lag* between the occurrence of events and the observed improvement of the QoI at the sink. As updated measurements arrive at the SN output, the pseudoinverse proves to be an effective means of determining the state of the Triggers.

4. Conclusions and Future Work

In this work we have used some practical examples to illustrate the concept of QoI at a SN output. We have observed how in this context, QoI is a time-varying quantity, influenced by sampling rate, network properties, and the nature and complexity of the scenario and monitored environment. By treating the environment as a series of video frames and interpreting the outputs in a way which aims to reconstruct these frames, the PSNR (a common metric used for image and video quality), was meaningfully applied to the problems as a QoI metric.

Future work will expand on the range of scenarios, number of experimental variables employed, and the algorithms used to process the outputs of the SN. For example the SN routing protocol used in the experiments may influence the QoI by impacting the transmission success rates. We also intend to use neural network pattern recognition techniques to address the problem presented in Section 3.2.

Acknowledgements Research was sponsored by the U.S. Army Research Laboratory and the U.K. Ministry of Defence and was accomplished under Agreement Number W911NF-06-3-0001. The views and conclusions contained in this document are those of the author(s) and should not be interpreted as representing the official policies, either expressed or implied, of the U.S. Army Research Laboratory, the U.S. Government, the U.K. Ministry of Defence or the U.K. Government. The U.S. and U.K. Governments are authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation hereon.

References

[1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci. Wireless sensor networks: a survey. *Computer Networks*, 38:393–422, 2002.

[2] C. Bisdikian. On sensor sampling and quality of information: A starting point. *Fifth IEEE International Conference on Pervasive Computing and Communications Workshops*, pages 279–284, March 2007.

[3] C. Bisdikian. Quality of information trade-offs in the detection of transient phenomena. *Proceedings of the SPIE Unattended Ground, Sea, and Air Sensor Technologies And Applications IX Conference*, April 2007.

[4] R. Penrose. On best approximate solution of linear matrix equations. *Proceedings of the Cambridge Philosophical Society*, pages 17–19, 1956.

[5] F. Ye, A. Chen, S. Lu, and L. Zhang. A scalable solution to minimum cost forwarding in large sensor networks. *Proceedings of the 10th International Conference on Computer Communications and Networks*, pages 304–309, October 2001.

[6] S. Zahedi and C. Bisdikian. A framework for qoi-inspired analysis for sensor network deployment planning. *2nd International Workshop on Performance Control in Wireless Sensor Networks*, October 2007.