Abstract—Distributed sensor systems composed of spatially distributed micro sensor nodes have been proposed for large scale monitoring applications. In these systems, nodes aggregate their sensor data to provide real time information about the underlying state. To extend the lifetime each node of the system has to limit the complexity of the sequential fusion algorithm. In this paper we derive optimal likelihood quantization rules for maximizing sequential detection performance. The resulting sequential detection algorithm is in the form of a finite state machine ideal for implementation in low complexity/low power devices.

Index Terms—Low power signal processing, Quantized Likelihood, Sequential tests with finite memory

I. INTRODUCTION

Spatially distributed set of micro sensor nodes can provide surveillance and monitoring of large scale structures. To detect and localize faults sensors have to integrate their information across time. Successful adoption of large scale distributed sensor systems is only possible if such networks can provide long lifetime. As a result temporal fusion of the sensor data has to be performed using low complexity/low power algorithms. Sequential decision procedures rely on computing, aggregating and communicating likelihood information at high precision which might be unsuitable for low power sensor nodes with limited computation and communication capability. Here, we consider design of quantized likelihood algorithms in the form of finite state machines suitable for implementation in low complexity devices.

Sequential aggregation rules with finite memory have been studied by Hellman and Cover [2], [3], [4]. They consider fixed sample size tests and derive quantization rules that are asymptotically optimal. Ertin [5] extended bounded memory technique to sequential tests and derived necessary equilibrium conditions for likelihood quantization. Varshney [6] consider the related problem of optimal quantization of prior information for fixed sample-size problems in Bayesian setting. In later work, they consider [7] extensions to distributed collaborative decision making. Varshney et al. interpreted their Bayesian prior quantization using human brain decision making example and derived the optimality condition for the quantizer.

In this paper, we provide necessary and sufficient conditions for optimal likelihood quantization for sequential testing.

Based on these results, we introduce an iterative algorithm based on policy iteration [8], which in each iteration improves the next step policy based on current state and has been proved to take less time to converge than value iteration. Our proposed algorithm converges to optimal quantization rules. We show that simple finite state machine decision rule with small number of states can approximate optimal sequential test performance closely. We study the performance of quantized likelihood decision rules as number of memory states and sampling cost is varied using simulation studies.

II. SEQUENTIAL PROBABILITY RATIO TEST

Sequential probability ratio test (SPRT) is a form of detection using repeated observations where the number of observations is an unknown random variable. Wald has shown in [1] that SPRT is superior in terms of the expected number of samples required to obtain the same probability of error as a fixed length test.

Let $y_0, y_1, \ldots$ be an infinite sequence of random variables in $\mathbb{R}^N$, conditionally independent and identically distributed with p.d.f. $f(y|H_1)$ (or equivalently with p.m.f. $Pr(Y|H_1)$), under either hypothesis $H_0$ or $H_1$. Further let $\pi$ be the prior probability of hypothesis $H_1$ and the likelihood ratio $\Lambda(y')$ is defined as $f(y|H_1)/f(y|H_0)$. In our study, the probability density function under both hypotheses $f(y|H_i)$ and the prior $\pi$ are assumed to be fully known. The sequential test is performed by a finite state machine with $L$ bits of memory ($M = 2^L$ states). At each state $l \in \{1, 2, \ldots, 2^L\}$ the decision rule is specified by the triple $\delta^l, \gamma^l, \eta^l$, where $\delta^l: \mathbb{R}^N \to \{0, 1\}$ is the stopping rule, $\gamma^l: \mathbb{R}^N \to \mathbb{R}$ is the final decision, and $\eta^l: \mathbb{R}^N \to \{0, 1\}$ is the state transition rule. If the detector is at state $l$ in the $t^{th}$ iteration, it computes $\delta^l(y^{t})$, $\gamma^l(y^{t})$, and $\eta^l(y^{t})$ using the most recent observation $y^{t}$. If $\delta^l(y^{t}) = 0$ the detector stops and declares a final decision using the result of $\gamma^l(y^{t})$, else if $\delta^l(y^{t}) = 1$ the detector jump to a new state for the next iteration based on $\eta^l(y^{t})$ and make a new measurement $y^{t+1}$. In this paper, we restrict our attention to non-randomized decision rules and consider the problem of minimizing the probability of error under constraints on the average number of observations and the memory size.
We start by analyzing the problem with no memory, and then extend our theory to the case with \( L \) bits of memory.

III. Optimal Sequential Probability Ratio Test without Memory

The simplest case in our study is to design a optimal detector for a sequential test without memory (i.e. with only one rejection state). In this case the state transition rule is trivial and we need to specify only the stopping rule \( \delta \) and the final decision rule \( \gamma \). Define three regions: \( \mathcal{H}_0 = \{ y^t \in \mathbb{R}^N | \delta(y^t) = 0 \text{ and } \gamma(y^t) = 0 \} \), \( \mathcal{H}_1 = \{ y^t \in \mathbb{R}^N | \delta(y^t) = 0 \text{ and } \gamma(y^t) = 1 \} \), and \( \mathcal{R} = \{ y^t \in \mathbb{R}^N | \delta(y^t) = 1 \} \). If \( y^t \) is in \( \mathcal{H}_i \), the detector stops and makes a decision of \( H_i \). If \( y^t \) is in \( \mathcal{R} \), the detector makes another observation.

For a given test \( \{ \delta, \gamma \} \), we can calculate the probability of miss detection \( P_M \), probability of false alarm \( P_F \), and expected number of samples \( N(H_i) \) under hypothesis \( H_i \) using:

\[
P_M = \frac{P(y^t \in \mathcal{H}_0 | H_1)}{1 - P(y^t \in \mathcal{R} | H_1)} \tag{1}
\]

\[
P_F = \frac{P(y^t \in \mathcal{H}_1 | H_0)}{1 - P(y^t \in \mathcal{R} | H_0)} \tag{2}
\]

\[
N(H_0) = \frac{1}{1 - P(y^t \in \mathcal{R} | H_0)} \tag{3}
\]

\[
N(H_1) = \frac{1}{1 - P(y^t \in \mathcal{R} | H_1)} \tag{4}
\]

We want to find the optimal test that minimizes the sum of expected observation cost and the probability of error given by:

\[
C = \pi P_M + (1 - \pi) P_F + c[(1 - \pi) N(H_0) + \pi N(H_1)] \tag{5}
\]

where \( c \) is the cost of each observation. This criterion can be extended trivially to include cross terms in Bayes decision cost and state dependent cost of experimentation.

**Theorem 1.** The optimal sequential test \( \{ \delta^*, \gamma^* \} \) with no memory is specified by two thresholds \( \{ \Lambda_1, \Lambda_2 \} \). The test is stopped and \( H_0 \) is declared if \( \Lambda(y^t) < \Lambda_1 \); the test is stopped and \( H_1 \) is declared if \( \Lambda(y^t) > \Lambda_2 \); and the test continues with the next sample otherwise.

**Proof:** First, to evaluate the condition for the optimal boundaries, we take the derivative of (5). Unfortunately, we cannot get an explicit expression for the boundary. But, an implicit relationship between the boundary and other parameters can be derived. We first differentiate (5) with respect to \( \Lambda_1 \), which is the likelihood ratio value of the lower bound, and then set that to zero, we get the following implicit condition:

\[
\Lambda_1 = \frac{1 - \pi}{\pi} \frac{N(H_0) P_F + c N^2(H_0)}{N(H_1) P_M - N(H_1) P_M - c N^2(H_1)} \tag{6}
\]

Since the state machine can not change its decision rule after we configure it, it will use the same decision rule throughout the test. Therefore, an optimal decision rule we choose from the beginning of the test should be exactly the same as if we are to determine it at any intermediate time slot. In other words, optimal decision rule is independent of time. Therefore, we can also model the test as described next.

For any intermediate iteration, the detector is faced with the same decision making problem. It has an expected cost-to-go function \( V(\pi(y^t)) \) and an expected cost of terminating test function \( U(\pi(y^t)) \) calculated using the observation value \( y^t \) of the current step:

\[
V(\pi(y^t)) = (1 - \pi(y^t))(cN(H_0) + P_F) + \pi(y^t)(cN(H_1) + P_M) \tag{7}
\]

\[
U(\pi(y^t)) = \min\{\pi(y^t), 1 - \pi(y^t)\} \tag{8}
\]

where \( \pi(y^t) \) is the a-posterior probability of \( H_1 \) after observing \( y^t \). Simply from the one step Bayes’ rule, we can calculate:

\[
\pi(y^t) = \frac{\tilde{\pi} \Lambda(y^t)}{\tilde{\pi} \Lambda(y^t) + 1 - \tilde{\pi}} \tag{9}
\]

where \( \tilde{\pi} \) is the prior of \( H_1 \) for this intermediate iteration problem. Since we mentioned before, the optimal decision rule should not change whether we are at the beginning of the test or at some intermediate time slot. We should choose \( \tilde{\pi} \) to guarantee that we will end up finding the identical decision rule.

To make the optimal decision rule of the same result, we take the intersection of (7) with (8) and evaluate the lower one using expression (6), then plug this into (9) we get:

\[
\tilde{\pi} = \frac{\pi N(H_1)}{\pi N(H_1) + (1 - \pi) N(H_0)} \tag{10}
\]

This choice of \( \tilde{\pi} \) can be interpreted as a quantized version of prior probability. A quantizer obtained a value from the posterior probability of the last time slot and quantize that value to give the prior of current time slot.

This result demonstrates a fact that, the posterior probability of one hypothesis under unknown time slot given the test is not terminated yet is equal to the proportion of number of visits to this rejection state under that hypothesis out of all expected number of visits to the rejection state for all possible hypotheses.

Since \( V(\pi(y^t)) \) is a linear function of \( \pi(y^t) \), it will have at most two intersection points with \( U(\pi(y^t)) \). Typically the number of intersection points are two, specifying three regions for the test. For situation where sampling cost is really high, which makes the agent being not affordable to even one observation, there will be no intersection point. Since \( \pi(y^t) \) is also a monotone transformation of the likelihood ratio \( \Lambda(y^t) \), the two intersection points of \( \pi(y^t) \) will also divide the region of \( \Lambda(y^t) \) into three parts, which proved our theorem.

IV. Optimal Sequential Probability Ratio Test with L Bits of Memory

For a sequential test with \( L \) bits of memory, the form of the test is similar to the test without memory, except for that the rejection region now needs to be further divided into \( 2^L \)
The optimal sequential test

Theorem 2. The optimal sequential test \( \{ \delta^*, \gamma^*, \eta^* \} \) with \( L \) bits of memory is specified by \( 2^L + 1 \) boundaries \( \{ \pi_b^m \}_{b=1}^{L} \) and \( 2^L \) centroids \( \{ \tilde{\pi}^j \}_{j=1}^{2^L} \). The test is stopped and \( H_0 \) is declared if \( \tilde{\pi}(y^t) < \pi_b^m \); the test is stopped and \( H_1 \) is declared if \( \tilde{\pi}(y^t) > \pi_b^m \); and the test continues with the \( m \th \) state if \( \pi_b^m < \tilde{\pi}(y^t) < \pi_b^m+1 \).

Proof: We first claim that the problem of optimizing for each initial state \( k \) the equation (11) is equivalent to the following problem under some specified choices of quantized prior probability. Consider the decision at state \( l \). Expected cost of continuing the test with state \( m \) is given as:

\[
V^m(\pi^l(y^t)) = \pi^l(y^t)(cN_k(H_0) + P^m_{\delta^*} + (1 - \pi^l(y^t)))(cN_k(H_0) + P^m_{\delta^*})
\]

and the expected cost of terminating test is given by:

\[
U(\pi_l(y^t)) = \min\{\pi_l(y^t), 1 - \pi_l(y^t)\}
\]

Each \( V^m(\pi^l(y^t)) \) is a linear function of \( \pi^l(y^t) \). Thus, the lower envelope of the linear function \( \{V^m(\cdot)\} \) is a concave function defined with \( 2^L - 1 \) intersections. The lower envelope \( \min_m\{V^m(\pi^l(y^t))\} \) can intersect with \( U(\pi_l(y^t)) \) at most two points, completing our proof.

Next, we prove the claim we made in transforming the problem. To guarantee the equivalence between the optimization problem of (11) and the optimization problem of (16), (17), we need to derive the expression of the optimal quantizer similar to the no memory case. We evaluate the first order derivative of the unconstrained problem of optimizing (11). The first order derivative of (11) can be calculated by using the first order derivatives of (12), (13) and (14), where the inverse transition matrix can be substituted using (15). Next, we evaluate the intersection of (16) and (17). For simplicity, we can choose the most left boundary \( b_1 \) to show the equivalence. By going through those procedures, we can get the following condition for the equivalence:

\[
\tilde{\pi}^j = \frac{\pi N_k^j(H_1)}{\pi N_k^j(H_1) + (1 - \pi) N_k^j(H_0)}
\]

Thus, by choosing such a series of \( \tilde{\pi}^j \), we can guarantee that the two optimization problems are giving exactly the same result.

Similar to the no memory case, the \( \tilde{\pi}^j \) can be interpreted as a quantized version of prior probabilities.

V. Simulation Result

We briefly introduce our algorithm based on the equivalent condition derived above. We arbitrarily pick initial set of quantizer centroids and boundaries but in ascending order to save computation. For each initial state, we calculate the abscissa using (16) and (17), and then move current boundary towards the calculated value by a small stepsizes. Next, we recalculate the centroid using (18). By going through those iterations, we terminate when all those boundaries are not changing any more (or change less than some threshold).

To show the effect of the size of memory and the effect of the optimization to the performance of the test, we calculate the optimal agents’ cost as the size of memory grows and compare those results to both the infinite size memory case and a non-optimally designed agent which is simply an equal spaced quantizer case.

The policy iteration algorithm works efficiently as the size of memory is not large. But, when the size of memory
relatively large, it will take a lot of computations to design the agent optimally. Therefore, it is necessary for a designer to know to what extend the optimal agent may outperform non-optimal agent as the memory size grows.

Here, we model the non-optimal agent as a simple equal space quantizer. Obviously, the performance of both optimal agent and non-optimal agent will become better as the memory size becomes larger and converge to the infinity memory size case. This limiting case cost can be calculated using the expression provided by Levy in [10], which gives the result of 0.0835. For the optimal agent, we implement our algorithm described above. Plotting them together, we get Fig.1 and Fig.2.

Fig. 1: Optimal Agent Compared with Non-optimal Agent

Fig. 2: Optimal Agent Compared with Continuous Case

Here, we may notice that optimal quantizer outperform equal spaced quantizer only when the memory size is very small. The graphs demonstrate that with three bits (8 states) of memory the performance of an optimally designed agent is almost the same as a non-optimal agent.

In Fig.3, we study the effect of step sampling cost, we let $c$ vary from 0.02 to 0.001. An obvious gap in performance is noticed between the optimal and non-optimal agent.

VI. CONCLUSION

In this paper, we study the case where memory size constraint is given to the sequential test. We formulate the problem as an optimal prior probability quantizer design under the sequential setting. Our result shows that the problem can be solved using dynamic programming by formulating the expected cost to continue and the expected cost to terminate as functions of the decision rules, in which setup the quantized prior probability for one hypothesis given the test is in a particular state can be proved to be equal to the proportion of number visiting that state given that hypothesis out of that of all possible states’.

ACKNOWLEDGMENT

This research was partially supported by Army Research Office grant W911NF-11-1-0391.

REFERENCES