ON THE EQUIVALENCE OF REGULAR GRAMMARS AND STOCHASTIC CONSTRAINTS: APPLICATIONS TO IMAGE PROCESSING ON MASSIVELY PARALLEL PROCESSORS

Michael I. Miller [†], Badrinath Roysam, Kurt Smith, Jan Tijmen Udding
Washington University
St. Louis, MO 63130

ABSTRACT

In this paper we propose a general method for mapping regular grammars to their equivalent stochastic representations, thereby allowing for a unified solution of stochastic estimation and or statistical pattern recognition problems over rule generated constraint spaces. For current problems in image processing in which features of the images are described by deterministic rules, and the measurements from which the images are reconstructed are samples of a probability distribution, such a synthesis is absolutely invaluable. The basic approach taken is to establish the formal connection of rules to Chomsky grammars, and to generalize the original work of Shannon on encoding rule-based channel sequences in Markov chains of the same entropy.

Coupling these results to the stochastic diffusions algorithms which sample these Gibbs distributions, yields the all important practical results that highly parallel computers may be used to sample the rule-constrained sets. We present results for image segmentation and reconstruction for emission tomography based on the DAP 510 mesh connected SIMD processor of Active Memory Technology.

[†] This work was supported via an NSF Presidential Young Investigator Award ECE-8552518 to Michael Miller and via the NIH Division of Research Resources grant RR01380.

1. Introduction

We are currently working on problems in image reconstruction and segmentation in which both rule-based constraints as well as stochastic priors are available expressing knowledge pertaining to the objects being imaged. One such example arises in emission tomography applications for which radioactive tracer doses are limited with the resulting images exhibiting rather severe speckle artifacts due to the independent increments property of the data-limited Poisson measurements. [1, 2]. We incorporate Good's roughness constraint, [3, 4], with this constraint controlled via an edge process for which there are various regular grammatical connectedness and curvature constraints. A second example arises in boundary tracing and segmentation for electron microscopy, [5], for which models in the form of the attribute grammars of Fu are available describing the small number of organelles being segmented.

This paper provides the starting point for a unified framework for incorporating deterministic rule based constraints into stochastic estimation problems. For the purpose of illustrating the issue of combining rule-based constraints into the stochastic estimation framework, consider the following straightforward statistical decoding problem. Assume that we have a binary message source in noise channel. Denote the input $n-$length code words to the channel as $\mathbf{x}_n = (x_1 x_2 \ldots x_n)$, and the output of the channel as $\mathbf{y}$. Then the minimum probability of error decoder is the well known maximum a posteriori (MAP) estimate of $\mathbf{x}_n$ given by

$$\hat{\mathbf{x}}_n = \arg\max_{\{\mathbf{x}_n\}} [\log P(\mathbf{y}|\mathbf{x}_n) + \log P(\mathbf{x}_n)] \,, \tag{1.1}$$

with $P(\mathbf{y}|\mathbf{x}_n)$ the probability density describing the noise source on the channel, and $P(\mathbf{x}_n)$ the probability distribution of the message source. The solution of (1.1) is conceptually straightforward, and is ideally suited for use of gradient descent [6, 7] and simulated annealing [8] methods. The problem explored in this paper is how the solution of (1.1) changes if the constraints on the message source are not in the form of a probability distribution, but rather in the form of rules corresponding to formal grammars. For some applications, this seems a more reasonable model for the source. One example explored in the next section is the high-level data link control (HDLC) language [9], where the messages on the channel satisfy simple run-length constraints. The fact that the prior information on the message source comes in the form of a set of deterministic rules suggests the decoding problem is a constrained optimization problem, where the constraint set (define it as $\mathbf{L}_n$) is the set of all n-length run constrained sequences, and the optimization problem is to maximize the probability of the measurement $\mathbf{y}$ with respect to $\mathbf{x}_n \in \mathbf{L}_n$. The constrained maximum-likelihood estimator (MLE) is given by

$$\hat{\mathbf{x}}_n = \arg\max_{\{\mathbf{x}_n : \mathbf{x}_n \in \mathbf{L}_n\}} [\log P(\mathbf{y}|\mathbf{x}_n)] \,. \tag{1.2}$$

While the constrained optimization problem of (1.2) is well defined, in general, the rules describing the constraint space may be fairly complicated. Newest approaches for maximizing the probability distribution $P(\mathbf{y}|\mathbf{x}_n)$ by gradient descent and annealing methods are not applicable as the constraint spaces corresponding to the language constraints must be incorporated.

2. Mapping regular grammars via maximum entropy Gibbs distributions

Having stated an example of a particular rule-based constraint set, we now follow Chomsky [10–12] and formalize the class of rules and the languages which they generate, and then demonstrate that these languages may be generated as samples of maximum entropy Gibbs distributions. As an intermediate step we first show how certain types of regular grammars correspond to Markov sources. We begin with two definitions.

Definition 1. A regular grammar $\mathbf{G}$ is a quadruple $\langle \mathbf{V}_N, \mathbf{V}_T, S_0, \mathbf{R}\rangle$ where $\mathbf{V}_N$ and $\mathbf{V}_T$ are finite sets of non-terminal and terminal symbols respectively, $S_0 \in \mathbf{V}_N$ is the start state and $\mathbf{R}$ is a finite set of production rules. The productions rules in $\mathbf{R}$ are independent of the context in which the substitutions to the non-terminal symbols are applied, and of the form

$$S_i \rightarrow W_j S_k \quad \text{or} \quad S_i \rightarrow W_j \quad \text{where} \quad W_j \in \mathbf{V}_T \quad \text{and} \quad S_i, S_k \in \mathbf{V}_N. \tag{2.1}$$

Each rule consists of the transformation of a non-terminal symbol to either a terminal followed by a non-terminal, or a terminal alone. The language $\mathbf{L}(G)$ generated by G is the set of all sequences consisting of terminal symbols that come about by starting in S_0 and subsequently applying a production rule to a non-terminal symbol until only terminal symbols are left. We denote the subset of $\mathbf{L}(G)$ of all strings of length n by $\mathbf{L}_n(G)$.

Definition 2. Define for these regular grammars the transition matrix $\mathbf{B}$ with entry $B(i,k)$ to be 1 if there is a production from S_i to S_k, and 0 otherwise.

For purpose of associating the ergodic finite state Markov sources with regular grammars, we impose the following additional restrictions upon the grammars. First we assume that no two arcs emanating from the same state have equal labels, and no two arcs from the same state lead to the same state. This implies that the sequence of of non-terminal symbols can be uniquely associated with a sequence of symbols in the language and vice versa. That is, for $S_i \rightarrow W_j S_k$ and $S_i \rightarrow W_l S_m$ then $(j = l)$ implies $(S_k = S_m)$. As proven in Chomsky and Miller [11] this does not restrict the class of regular grammars. The second requirement upon our regular languages are that they are prefix-closed, i.e., $S_i \rightarrow W_j S_k$ is a production rule then so is $S_i \rightarrow W_j$. Our third assumption is that the state graph of the grammar is irreducible so that any non-terminal symbol can be reached from any other non-terminal, and aperiodic so that the lengths of the loops beginning and ending in any of the states S_i have gcd's 1. Given these assumptions we now define the finite state Markov sources

Definition 3. Consider the irreducible, aperiodic finite state Markov sources to be a finite directed graph, with nodes of the graph corresponding to states and arcs labeled with a source symbol from the terminal symbol alphabet. Associated with each state is a probability $Q(S_i, S_k)$ from $\mathbf{Q}$ describing the probability of going from state S_i to state S_j. We define the language of n-length sequences $\mathbf{L}_{n,\varepsilon}(M)$ generated by the Markov source as the set of all strings $\mathbf{x}_n$ with normalized log-probability bounded below by $-H - \varepsilon$, for ε a finite constant greater than zero, and H the entropy of the Markov source.[1] That is $\mathbf{L}_{n,\varepsilon}(M) = \{\mathbf{x}_n : 1/n \log P(\mathbf{x}_n) > -H - \varepsilon\}$.

[1] The entropy per symbol is defined as $H = \lim_{n\rightarrow\infty} -\frac{1}{n}\sum_{\mathbf{x}_n} P(\mathbf{x}_n)\log P(\mathbf{x}_n)$.

Definition 4. Also define the state of string $\mathbf{x}$ in $L(M)$, denoted by *state*$(\mathbf{x})$, to be the state that the generation of $\mathbf{x}$ starting in S_0 leads to.

While it is clear how to associate a regular grammar **G** in this class to a finite state Markov source of equal generative power, we do not yet have an explicit mechanism for choosing the Markov chain probabilities associated with each production rule. There is an infinite family of probabilistic finite-state sources which respect the syntax of **G**, yet there is a unique set of production rules which generate a language of identical size. By properly choosing the transition probabilities so as to maximize entropy and satisfy the production rules it follows that the finite-state language and the regular language generated by the production matrix **B** are both "probabilistically" as well as "structurally" equivalent. The choice of production rule probabilities Grenander [13] has termed the ***style*** of the grammar.

Recognizing issues of a similar kind for specifying constrained channels (which are particular examples of regular grammars), Shannon [14] described the generation of the unique Markov chains satisfying a given set of channel constraints and maximizing entropy (see Appendix 4 of [14]). The construction we now state as Theorem 1 follows his results.

Theorem 1. *Let the finite state ergodic source* $\mathbf{M}$ *have transition probability matrix* $\mathbf{Q}$ *with ikth entry* $Q(i,k) = \frac{B(i,k)e(k)}{\lambda e(i)}$, *where* $\mathbf{e}$ *is the right eigenvector corresponding to the unique largest eigenvalue* λ *of the production matrix* $\mathbf{B}$ *associated with the regular grammar.* [2] *Then, the set of* $n-$*length sequences* $\mathbf{L}_{n,\varepsilon}(M)$ *produced by the Markov source is probabilistically and structurally equivalent to the corresponding regular language* $\mathbf{L}_n(G)$ *in the following senses.*

(i) The probabilistic equivalence corresponds to $P(\mathbf{x}_n)$ converging to uniformity with Probability 1 (denoted as $\overset{a.e.}{\rightarrow}$), for all $\mathbf{x}_n \in \mathbf{L}_{n,\varepsilon}(M)$ and large n:

$$\frac{1}{n}\log P(\mathbf{x}_n) \overset{a.e.}{\rightarrow} -H \quad \text{for } \mathbf{x}_n \in \mathbf{L}_{n,\varepsilon}(M) \text{ as } n \rightarrow \infty, \tag{2.2}$$

where H is the entropy per source symbol.

(ii)The structural equivalence follows from the fact that the finite state source sequences are legal derivations in the regular grammar and the normalized rate of exponential growth of the grammar and the Markov generated languages are equal. That is for all $\varepsilon > 0$,

$$\lim_{n\rightarrow\infty} \frac{1}{n}\log|L_{n,\varepsilon}(M)| = \lim_{n\rightarrow\infty}\frac{1}{n}|L_n(G)| = H = \log\lambda. \tag{2.3}$$

Proof. That the languages are probabilistically equivalent according to (i) follows from the ergodic property of the chain. That is for any measurable function $f_k(\mathbf{x}_m)$ of a finite

[2] That there is a unique largest eigenvalue follows from theorems of Frobenius on positive matrices and the assumption of connectedness of the graph.

set k of variables $\{x_m, x_{m-1}, \ldots, x_{m-k}\}$, then

$$\frac{1}{n}\sum_{m=k}^{n} f_k(\mathbf{x}_m) \overset{a.e.}{\rightarrow} E\{f_k(\mathbf{x}_m)\} \quad \text{as } n \rightarrow \infty, \tag{2.4}$$

with the expectation $E\{\ \}$ taken with respect to the measure of the Markov chain. Since $\log P(\mathbf{x}_n) = \sum_{i=1}^{n} \log Q(S_i, S_{i+1})$ and all states S_i are a function of a finite number of random variables, the almost everywhere convergence follows.

The structural equivalence of (ii) has to do with the facts that sequences generated via the Markov source are legal derivations of the regular grammar and the number of sequences in the two languages are identical. That the Markov sequences are legal derivation of the grammar follows simply from the fact that the Markov chain has non-zero transition probabilities only for state pairs corresponding to production rules in the regular grammar. Now, the exponential rate of growth of the language is given by [11, 14]

$$\lim_{n\rightarrow\infty} \frac{1}{n}|L_n(G)| = \log\lambda, \tag{2.5}$$

with λ corresponding to the largest eigenvalue of the production matrix $\mathbf{B}$. This results from the growth rate being determined by the entries in the nth power of the production rule matrix $\mathbf{B}_n$, which grow as the largest eigenvalue λ of $\mathbf{B}$ (see [14] or [11]). Since as is well known from the Shannon-McMillan theorem the logarithmic growth of the domain of the Markov chain is given by the entropy H, we must show that the entropy H equals $\log\lambda$. This involves a straightforward calculation of H according to the definition of the entropy of the Markov chain [15] given by

$$H = -\sum_i q(i)\sum_j Q(i,j)\log Q(i,j), \tag{2.6}$$

where $\mathbf{q}$ is the limit distribution of the Markov chain corresponding to the left eigenvector of $\mathbf{Q}$ with eigenvalue 1. Using

$$Q(i,k) = \frac{B(i,k)e(k)}{\lambda e(i)}, \tag{2.7}$$

and substituting into (2.6) yields $H = \log\lambda$, proving part (ii).

We now explore a method for sampling the constraint spaces corresponding to languages in parallel. Both the regular and finite-state languages are generated using either sequential application of the rules or sequential simulation of the Markov distribution. Our goal is to sample each n-length sequence $\mathbf{x}_n \in \mathbf{L}_n$ simultaneously using n processors, with each processor generating one of the n symbols of a particular message $\mathbf{x}_n$ in parallel! By rewriting the Markov probabilities as Gibbs' distributions, thereby releasing the causality constraint inherent to Markov chain states, a direct method for the parallel computation of entire sequences in the language becomes possible. We proceed by stating an approach, first heralded by Jaynes [16] as a general principle for generating Gibbs' distributions subject to mean-value constraints which maximize entropy. This allows us to generalize Theorem 1 maximizing entropy subject to particular regular grammar constraints to more general constraint rules and languages associated with them. As argued by Jaynes, the principle of maximum entropy generates the distribution representing some specified set of mean-value constraints [17], without adding additional constraints. In this sense it is the analogue of Theorem 1 and the original approach of Shannon for constrained channel sequences.

Theorem 2. *The distribution maximizing entropy subject to the mean-value constraints*

$$E\{o_j(\mathrm{x}_n)\} = O_j \quad \text{for } 1 \le j \le M, \tag{2.8}$$

is given by

$$P(\mathrm{x}_n) = 1/Z_n \exp\left[\sum_{j=1}^{M} \alpha_j o_j(\mathrm{x}_n)\right], \tag{2.9}$$

where α_j are Lagrange multipliers chosen to satisfy the constraints, and Z_n normalizes $P(\mathrm{x}_n)$ to have measure 1.

Proof. The distribution $P(\mathrm{x}_n)$ given by (2.9) results by maximizing the entropy functional $-\sum_{\mathrm{x}_n} p(\mathrm{x}_n)\log p(\mathrm{x}_n)$, with the uniqueness of (2.9) resulting from the strict concavity of the entropy functional.

It follows that the class of Markov chains and their corresponding regular grammars are maxent distributions subject to mean-value constraints on functions of the type (2.8). We first extend the state graph corresponding to the Markov source M with one additional state ϕ, called undefined. We add from any node to ϕ the arcs labeled with the symbols in the alphabet of M not emanating from that node in the original state graph. The probabilities associated with these new arcs are 0. Hence trying to generate a string not in the language leads to the state ϕ.

Definition 5. Define the characteristic function on the ordered pair of states as follows:

$$\begin{aligned} I_{(i,k)}(S_m, S_n) &= 1 \quad \text{for } S_m = i, S_n = k; \\ I_{(i,k)}(S_m, S_n) &= 0 \quad \text{otherwise.} \end{aligned} \tag{2.10}$$

Now we prove the following corollary.

Corollary 1. *Let a finite state Markov source* M *with stochastic matrix* **Q** *have transition probabilities $Q(i,k)$. Then the set of $n-$length sequences $\mathrm{L}_{n,\epsilon}(M)$ corresponding to the Markov source may be generated by sampling the Gibbs' measure $P(\mathrm{x}_n)$ given by*

$$P(\mathrm{x}_n) = \exp\left[\sum_{i,k} \alpha_{i,k} \sum_{m=1}^{n} I_{i,k}(state(\mathrm{x}_{m-1}), state(\mathrm{x}_m))\right], \tag{2.11}$$

with the constants $\alpha_{i,k}$ specified as follows:

$$\alpha_{i,k} = \log Q(i,k) \quad \text{for } Q(i,k) > 0; \tag{2.12}$$

$$e^{\alpha_{i,k}} = 0 \quad \text{for } Q(i,k) = 0. \tag{2.13}$$

Proof. All that needs be proven is that the Gibbs measure $P(\mathbf{x}_n)$ of (2.11) and the Markov chain probabilities are equal for all $\mathbf{x}_n \in \mathbf{L}_{n,\varepsilon}(M)$. This follows directly from the fact that for each $\mathbf{x}_n \in \mathbf{L}_{n,\varepsilon}(M)$, there is a unique state sequence $(S_{c_0}, S_{c_1}, \ldots, S_{c_n})$. Therefore, for each m in the sum of the Gibbs' distribution only one characteristic function is non-zero, yielding the distribution $P(\mathbf{x}_n) = \prod_{m=1}^{n} \exp[\alpha_{c_{m-1},c_m}]$. Since the α's have been chosen according to the conditional Markov chain probabilities, the Gibbs measure is $P(\mathbf{x}_n) = \prod_{m=1}^{n} Q(S_{c_{m-1}}, S_{c_m})$, which is precisely the Markov chain probability of message $\mathbf{x}_n$ with the associated state sequence $(S_{c_0}, S_{c_1}, \ldots, S_{c_n})$.

Before concluding this section we define the finite complement languages, as they play an important role in the embedding of certain kinds of constraints via Gibbs' distributions. Various of our edge grammars involve the complement grammar representation. To avoid the introduction of a new set of notation, we define them informally while alluding to the formal development in Chomsky and Miller. [11].

Definition 6. Informally, the complement grammar $\mathbf{G}^c$ of $\mathbf{G}$ is constructed by adding one new state to the original grammar, call it ϕ from the construction of Definition 4, to which all disallowed arcs from the original grammar $\mathbf{G}$ lead. For the sequences generated by $\mathbf{G}^c$, once the new added state is entered all possible sequences are allowed, and the process remains in the added state. As proven in [11] the complement grammar $\mathbf{G}^c$ has the property that its language $\mathbf{L}(G^c)$ is the complement of the original language $\mathbf{L}(G)$. That is, defining $\mathbf{U}$ to be the universal language of all strings which can be generated from the finite alphabet $\mathbf{V}_T$, then $\mathbf{U} = \mathbf{L}(G) \cup \mathbf{L}(G^c)$, with $\mathbf{L}(G)$ and $\mathbf{L}(G^c)$ having a null intersection.

The importance of this relationship is that in many instances, the regular language may be more simply generated by generating the complement of the complement language, i.e. $\mathbf{L}(G) = [\mathbf{L}(G^c)]^c$. This allows us to state the following proposition which constructs $\mathbf{L}(G)$ via a Gibbs' distribution generating the complement of the complement language. We do this by constructing the measure P with the property that it is zero for $\forall \mathbf{x} \in \mathbf{L}(G^c)$, and represents the largest language (maximum entropy).

Proposition. *Let $\mathbf{G}^c$ be the complement grammar of $\mathbf{G}$ with the set $\Phi = \{i : (i,j,\phi)\}$ being the set of states from $\mathbf{G}$ with disallowed arcs to ϕ. Then the distribution P maximizing entropy subject to the constraints*

$$E\{I_{i,\phi}(state(\mathbf{x}_{m-1}), state(\mathbf{x}_m))\} = 0, \quad \forall i \in \Phi,\ 1 \le m \le n, \tag{2.14}$$

generates the regular language $\mathbf{L}_n(G)$.

3. Regular language example

We proceed by illustrating these results via the 4-0,1 HDLC example posed in the introduction; that is the four symbol run-length constraint sequences of Section 1 have the set of seven non-terminals $\mathbf{V}_N \triangleq \{S_0 = \varepsilon, S_1 = 000, S_2 = 00, S_3 = 0, S_4 = 1, S_5 = 11, S_6 = 111\}$, and the terminals $V_T \triangleq \{0,1\}$, with the set of production rules given as

$$\mathbf{R} \triangleq \{S_0 \to 0S_3, S_0 \to 1S_4, S_1 \to 1S_4, S_2 \to 1S_4, S_2 \to 0S_1, S_3 \to 1S_4, \tag{3.1}$$
$$S_3 \to 0S_2, S_4 \to 1S_5, S_4 \to 0S_3, S_5 \to 1S_6, S_5 \to 0S_3, S_6 \to 0S_3\}.$$

The state S_0 is not reachable from any other state and is therefore not part of the irreducible set of states. Since the capacity of the language is solely determined by the rule matrix **B** defined over the irreducible set (see [11] for proof), we confine our considerations to the matrix **B** constructed from derivations from non-terminal S_i to S_k for $1 \leq i, k \leq 6$:

$$\mathbf{B} = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix} . \tag{3.2}$$

For the matrix **B** of (3.2), the exponential rate of growth of the language is given by the log of the largest eigenvalue $\log \lambda = 0.879416$. Applying Theorem 1 for the generation of the finite-state, stochastic representation of the regular grammar, the Markov chain defined via the following transition probability matrix **Q** generates a structurally equivalent language:

$$\mathbf{Q} = \begin{bmatrix} 0 & 0 & 0 & 1.0 & 0 & 0 \\ 0.3522 & 0 & 0 & 0.6478 & 0 & 0 \\ 0 & 0.4563 & 0 & 0.5437 & 0 & 0 \\ 0 & 0 & 0.5437 & 0 & 0.4563 & 0 \\ 0 & 0 & 0.6478 & 0 & 0 & 0.3522 \\ 0 & 0 & 1.0 & 0 & 0 & 0 \end{bmatrix} . \tag{3.3}$$

Applying the definition of the entropy of the Markov chain as

$$H = -\sum_i q(i) \sum_j Q(i,j) \log Q(i,j), \tag{3.4}$$

with **q** the left eigenvector of **Q** of (3.3) yields the entropy $H = 0.879416$. Note, since there is only one irreducible set of states we can choose any probabilities $p, 1-p$ for the productions $S_0 \xrightarrow{p} 0S_3$ and $S_0 \xrightarrow{1-p} 1S_4$, without altering the entropy of the Markov chain.

Returning to the original decoding problem stated in the introduction, it follows that the MAP solution requiring maximization of $P(\mathbf{y}|\mathbf{x}_n)$ with additive probability measure $P(\mathbf{x}_n)$ describing the message sequence is absolutely equivalent to generating the MLE over the constraint region $\mathbf{L}_n$. For the 4-0,1 constraint, $\log P(\mathbf{x}_n) \approx -n.0879146$ for $\mathbf{x}_n \in \mathbf{L}_n$, implying that for large n performing the MAP decoder of (1.1) and the constrained MLE of (1.2) are equivalent.

4. Parallel implementation

Conventional sampling methods such as the Metropolis algorithm offer a limited amount of parallelism in that sites in non-overlapping neighborhoods may be updated simultaneously. In image reconstruction for emission tomography [18] the neighborhoods correspond to line integrals through the image, thereby decreasing substantially the possible parallelism. This has led us to explore an alternate method based on the following Langevin stochastic differential equation:

$$d\mathbf{x}_n(t) = -\nabla E(\mathbf{x}_n)dt + \sqrt{2}d\mathbf{w}_n(t) \tag{4.1}$$

in which $E(\mathbf{x}_n)$ is the Gibbs energy and $\mathbf{w}_n$ is a standard vector Wiener process. It is well known that under certain regularity conditions [19–21] the stochastic diffusion has the following stationary density:

$$p(\mathbf{x}_n) = \frac{1}{Z}\exp[-E(\mathbf{x}_n)]. \tag{4.2}$$

In order to use (4.2) which generates a continuous-valued diffusion over $\mathbf{R}^n$ for generating sequences corresponding to a finite alphabet we must approximate the Gibbs' distributions defined on finite-valued domains via distributions defined over continuous ones. One approach outlined by Vichniac [22] for binary processes is to add a penalty term to the energy function which forces the diffusion to concentrate on the finite alphabet. The method adopted here is fundamentally different as we have had difficulties with the aforementioned approach since the large energy terms restricting the alphabet symbols overtake the gradient of the energy in (4.1). Instead we define a continuous random variable $\mathbf{u}_n \in \mathbf{R}^n$ over which Langevin's equation is applied, and then map $\mathbf{u}_n$ via a function f to the random vector $\mathbf{x}_n$ which is concentrated over the finite alphabet. By crafting the gradient in Langevin's equation on $\mathbf{u}_n$ properly, the resulting measure on $\mathbf{x}_n$ is a "good" continuous valued approximation to the desired distribution.

We proceed as follows. For simplicity define x' to be a scalar random variable on the alphabet $\{0,1\}$ with $E(x')$ the Gibbs' energy function. The corresponding Gibbs' distribution is given by

$$P(x'=0) = \frac{\exp[-E(0)]}{\exp[-E(0)]+\exp[-E(1)]}, \tag{4.3a}$$

$$P(x'=1) = \frac{\exp[-E(1)]}{\exp[-E(0)]+\exp[-E(1)]}. \tag{4.3b}$$

Now define the continuous random variables $u \in \mathbf{R}^1$ and $x \in (0,1)$ with the autologistic function f_α mapping u to x as follows:

$$x = f_\alpha(u) = \frac{1}{1+\exp[-\alpha u]}. \tag{4.4}$$

Note, by increasing the parameter α from 0 to ∞ the function f_α converges to a step function. The stochastic differential equation we define on u is given by

$$du(t) = -\frac{dE(x)}{dx}\frac{df_\alpha(u)}{du}\,dt + dw(t), \tag{4.5}$$

with E the identical energy function as defined in (4.2). It is now straightforward to show that for α large, the probability of x concentrates around the binary values $0, 1$ with the required probability of (4.2) [23]. We emphasize that the above approach may be simply extended to vector processes, and is most naturally implemented via analog circuits since the autologistic function is simply generated in hardware.

To illustrate the application of this method for representing the regular grammars, we have generated samples from the HDLC example of (3.1) using the diffusion based approach, concentrated over the binary symbol alphabet. For this case, since the number of states with disallowed arcs ($S_1 = 000, S_6 = 111$) is smaller than the number of states in the transition matrix corresponding to (3.3), we represent the language using the complement of the complement grammar. We proceed by simply embedding the most obvious constraint that the probability of there being a sequence with 4-0's or 4-1's is 0. The probability measure disallowing arcs from states 000, 111 to state $\phi_1 \triangleq 0000$, $\phi_2 \triangleq 1111$ and generating the language is given as follows:

$$P(\mathbf{x}_n) = 1/Z_n \exp\left[\sum_{k=1}^{n} \alpha[I_{\phi_1}(state(\mathbf{x}_k)) + I_{\phi_2}(state(\mathbf{x}_k))]\right], \tag{4.6}$$

with $\alpha \ll 0$. Since the indicator function on the state $I_\phi(\ )$ is equivalent to the logical and of four consecutive symbols of the same binary value, it may be rewritten as

$$P(\mathbf{x}_n) = 1/Z_n \exp\left[\alpha \sum_{k=4}^{n} (x_{n-3}x_{n-2}x_{n-1}x_n + \overline{x}_{n-3}\overline{x}_{n-2}\overline{x}_{n-1}\overline{x}_n)\right], \tag{4.7}$$

where $\overline{x}$ denotes complement of the binary symbol x. Now sampling from the distribution of (4.7) using the stochastic differential equation approach yields the following sample transition matrix $\widehat{\mathbf{Q}}$:

$$\widehat{\mathbf{Q}} = \begin{bmatrix} 0 & 0 & 0 & 1.0 & 0 & 0 \\ 0.3456 & 0 & 0 & 0.6544 & 0 & 0 \\ 0 & 0.4420 & 0 & 0.5580 & 0 & 0 \\ 0 & 0 & 0.5540 & 0 & 0.4460 & 0 \\ 0 & 0 & 0.6606 & 0 & 0 & 0.3394 \\ 0 & 0 & 1.0 & 0 & 0 & 0 \end{bmatrix}. \tag{4.8}$$

Comparing $\widehat{\mathbf{Q}}$ of (4.8) to that of (3.3) shows that the diffusion approach yields a fairly good continuous approximation to the original distribution.

5. A hierarchy for image processing

We now present a hierarchical approach to the image processing problem, parallel to that first proposed by Baker [24] for speech recognition, and one which incorporates the more recent attribute grammar approaches of Fu [25, 26] and others on hierarchies of controlling grammars [27]. We imagine that there are multiple grammars and stochastic models, organized into a hierarchy of probabilistic Gibbs' distribution representations. At each level of the hierarchy there is probabilistic data from which estimates of features of the image must be generated, with both grammatical as well as stochastic constraints. The power introduced via a hierarchical description is that the language theoretic generative power of the model may be increased, while maintaining relatively simple complexity at each level. For example, regular grammars controlling context-free grammars can generate context-sensitive languages [26–28].

The importance of the representation theorems of the regular grammars in this context is that grammar generated language constraints may be embedded within the stochastic estimation framework in a unified manner. To illustrate the hierarchy for our tomographic imaging applications [1, 29, 30] *at the lowest level 1* of the hierarchy the features correspond to pixel intensities within anatomic regions. For the tomography model, we assume that the measurements are Poisson distributed with means determined by generalized line-integrals through the underlying density of radioactive tracer (see Snyder et al. [18], this issue). Unfortunately, low radioactive tracer concentrations result in maximum-likelihood reconstructions which are noisy [2, 31]. We make the estimation of the pixel intensities robust via the introduction of Good's roughness prior [4]. *At level 2* we introduce a layer consisting of *image characteristic function attributes* **C** that serve to attribute the membership of pixels to different object types. These labels essentially serve to select the statistical models and priors applicable for the underlying pixel models in layer 1, i.e. the parameters in Good's roughness model for example. We estimate the image characteristic functions using hypothesis testing against different object region models. As a prior on these attributed characteristic functions we constrain them to be continuous over the interiors of object regions by inducing run-length type grammatical constraints, which we impose via the complement grammar representation. *At level 3 in the hierarchy*, we enforce the constraint that different regions in the image may be parts of different structures and should be smoothed accordingly. To accomplish this, Good's roughness is induced on level 1 over the interiors of regions by introducing a controlling set of edge sites. Weak configurational constraints on the edge site clusters are induced, such as ones that rule out the existence of parallel edge sites across a single pixel. To make the estimation of the edges robust, *we are presently working* on level 4 of the hierarchy where the edge sites are considered part of the "primal sketch" of Marr and Hildreth; they are therefore smooth (reasonable curvatures) and connected. These constraints we embed via simple "north-south" turning grammars on the edge sites. They are in turn attributed (in the sense of Fu) *level 5*, via hypothesis testing against various edge-site models which may be available.

To our knowledge the characteristic functions go back to 1977 with Nahi and Jahanshahi [32] who formulated the boundary estimation problem in terms of object characteristic functions. More recently they have been used by Derin et al. [33], and most recently Geman et al. [34]. The characteristic function formulation presented here and viewed as attributes of the pixels has been inspired by the work of Fu [25]. The notion of an independent edge site array for the acquisition of boundaries was proposed by Martelli [35] and heralded by Geman and Geman [8]. The edge sites serve as a controlling grammar, much like the punctuation process first introduced by Chomsky and Miller [11], on the lower-level characteristic function grammar as well as on Goods' roughness and corresponds to the phoneme boundaries in Baker's original Dragon System speech recognizer [24]. The notion of attributing clusters of edge-sites to particular models is precisely the approach taken by Tsai and Fu [25] in which the edge-site models correspond to the four different kinds of boundary segments of the various machine tools being recognized.

5.1 Segmentation

To illustrate the use of the attribute layer and the above controlling edge-sites we first explore a relatively simple estimation and segmentation problem based on the above hierarchy. For purpose of simplicity assume that we have a 1-dimensional estimation problem, with two model types $\mathbf{M}^0$ and $\mathbf{M}^1$ governing the image data, with each a Poisson counting process having intensity μ^0 and μ^1, respectively. The model in pixel i becomes

$$1/Z \exp[-\mu_i^k + \log(\mu_i^k)N_i], \tag{5.1}$$

with Z a normalizing constant and N_i the number of counts in pixel i. The segmentation problem is to determine which of the two models a pixel is associated with, and where the boundaries of the region are.

Now the attribute characteristic function layer **C** is determined via hypothesis testing against the aforementioned models. Defining hypothesis H^0 as occurring in pixel i when $C_i = 0$, and hypothesis H^1 as occurring when $C_i = 1$, this yields the following comparison of Bayes log-posteriors for choosing the attribute labels:

$$-\mu_i^1 + \log(\mu_i^1)N_i + \log P(C_i = 1) \underset{H_0}{\overset{H_1}{\gtrless}} -\mu_i^0 + \log(\mu_i^0)N_i + \log P(C_i = 0), \tag{5.2}$$

where $P(C_i = k)$ is a prior on choosing model $k = 0, 1$ in pixel i. We note, that for the actual implementation of the hypothesis test we use fully parallel gradient descent with respect to the attributes C_i. That is, following our approach in Section 4 we force $C_i \in (0, 1)$, concentrated near the binary values $0, 1$, and descend the following Gibbs energy with respect to $\mathbf{C}$:

$$\sum_i \left[C_i(-\mu_i^1 + \log[\mu_i^1]N_i) + (1 - C_i)(-\mu_i^0 + \log[\mu_i^0]N_i) + \log P(C_i)\right]. \tag{5.3}$$

Minimizing (5.3) with respect to the C_i's is a continuous approximation to the hypothesis test of (5.2).

Now the prior $P(C_i)$ correspond to two simple grammars expressing continuity on the characteristic function attributes, which are controlled by the edge sites. That is if in pixel i, hypothesis H^k is chosen with $C_i = k$, then grammar k produces symbol $C_i + 1 = k$. The edge sites control the characteristic function layer in precisely the same way as they control the pixel intensity interactions in previous papers [8, 35]. That is, the constraints are removed when straddling an active edge site.

Define the edge sites for the 1-dimensional case via the binary process e_i, where $e_i = 1$ if the site between pixels i and $i + 1$ contains the edge site; $e_i = 0$ otherwise. Then the edge sites are placed based on hypothesis testing using the attribute layer $\mathbf{C}$ as the data. That is, edge sites are placed at locations where the sharp discontinuity in the characteristic functions C_i is greater than some threshold. The hypothesis test is actually computed by doing gradient descent with respect to the edges e_i on the following Gibbs energy:

$$\sum_i (C_{i+1} - C_i)^2(1 - e_i) + \log P(e_i). \tag{5.4}$$

The prior $P(e_i)$ on the edge sites is a simple $2 - 1$ run-length constraint (as previously described) that forbids the formation of consecutive parallel edge-sites (edges straddling one pixel). Note well, in the above approach the edge-sites are placed based on discontinuities in the attribute function layer, not the pixel intensity values.

We have implemented via finite-differences on the AMT DAP500 parallel processor the gradient descent corresponding to the two layers of (5.3,5.4). Shown in Figure 1 are the results of applying this segmentation approach to a Poisson image. The top row shows the original 32 x 32 object (left column), with the Poisson data (right column). The intensities for the two models were chosen to have a moderately high signal-to-noise ratio (SNR) (low speckle). The bottom row shows the attributes (left column) and edge sites (right column) that were estimated using gradient descent of (5.3,5.4). The attributes and edges $\mathbf{C}, \mathbf{E}$ are real numbers between 0 and 1, with 1 corresponding to the brightest intensity. We emphasize that the $\mathbf{C}$ and $\mathbf{E}$ layers were computed jointly with every element of the $\mathbf{C}$ and the $\mathbf{E}$ layers updated synchronously in parallel.

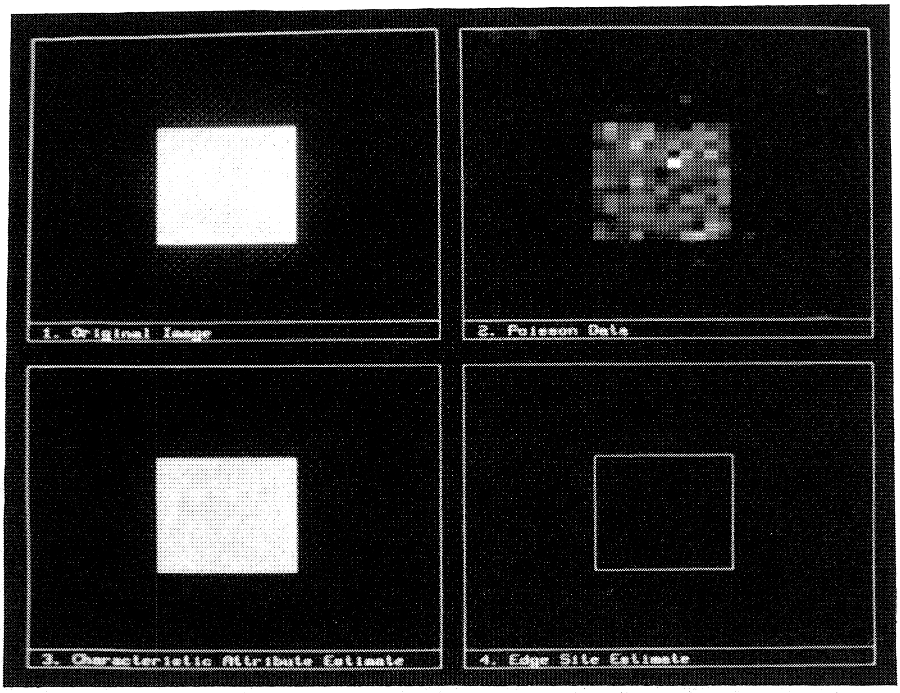

Figure 1: Top left shows the original two object image, with top right showing the Poisson data. Bottom left shows the values of the characteristic function, with white meaning object 1, and black object 2. Bottom right shows the edge sites generated from the characteristic layer.

Now returning to the placement of the edge sites in (5.4), we note a strong similarity to the original edge operator of Marr and Hildreth [36]. Ignoring for a moment the prior $P(e_i)$ in (5.4), the gradient descent algorithm places the edges at locations i where the function $(C_{i+1} - C_i)^2$ is a maximum, thereby minimizing (5.4). This corresponds precisely to the places where the Laplacian of the characteristic layer is zero. The crucial difference between our approach and the Marr-Hildreth operator for edge detection, is that the Laplacian would be computed on the characteristic function attribute layer, not the pixel intensity layer. It does, however, seem clear that if the underlying object models have no texture (as in our case where we have assumed two constant intensities μ^0, μ^1), the sufficient statistic upon which the hypothesis test is performed is simply the pixel intensities N_i. Therefore, for the simple textureless case at high SNR levels, we might expect that the placement of the edge-sites based on the Laplacian of the pixel intensity data should be similar to that derived using our attribute layer. Shown in Figure 2 are the result of comparing the edge sites generated with the attribute layer according to gradient descent on (5.3,5.4) and the Marr-Hildreth operator. The top row shows the results of a higher intensity Poisson image than that seen in Figure 1 (less speckle). The left column shows the Poisson data, the middle column the edges generated using the attribute layer and equation (5.4) to set the edge sites, and the right column the edge sites generated with the Marr-Hildreth operator; that is the Poisson data is first smoothed and then the zero crossings of the Laplacian are computed for placement of the edge sites. We see for this high intensity (high SNR) data that there is a fairly good correspondence between the edges placed at the boundary of the square phantom using both methods. The bottom row shows the results of segmentation from

Poisson data at a lower intensity. For the highly speckled low SNR Poisson data, the placement of the edges based on the Marr-Hildreth operator degenerates rapidly. Notice for this low intensity data the edges are no longer perfectly placed using our new method (middle column). However, we are presently working on higher levels which incorporate connectedness and curvature constraints on the edge sites.

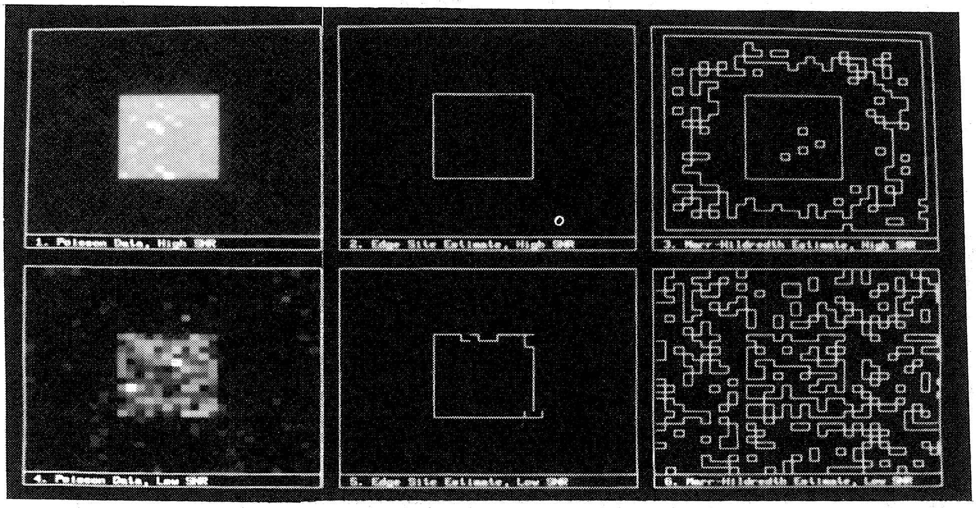

Figure 2: Top left shows the Poisson data, middle shows the edge site estimates generated from the attribute layer, and right the edge sites placed using the Marr-Hildreth operator. Bottom row is identical to the top row, simply generated from the lower SNR data.

5.2 Tomographic reconstruction

We have begun applying these ideas to the positron emission tomography (PET) problem. The PET model we adopt is described in greater detail in another article in this proceedings [18]. We assume that the measurements $M_\theta(j)$ form a Poisson counting process with mean $\mu_\theta(j) = \sum_i p_\theta(j|i)\lambda(i)$ for $1 \leq \theta \leq N_\theta$. For the Super PETT-I tomograph at Washington University, $N_\theta = 96$ view angles of data are measured with the point-spreads $p_\theta(\ |\ )$ corresponding to Gaussian-weighted surfaces directed along the line-of-flight angle θ, with full-width at half-maxima along the line-of-flight equal to 7.5 cm, and 1.0 cm perpendicular to the line-of-flight. The measurements are described by the Poisson log-likelihood given by

$$L(\lambda) = -\sum_\theta \sum_j \mu_\theta(j) + \sum_\theta \sum_j \log[\mu_\theta(j)] M_\theta(j), \tag{5.5}$$

and the estimation problem is to estimate $\lambda(i)$. It is well known that unconstrained nonparametric maximum-likelihood (ML) estimators may be fundamentally inconsistent [3, 37, 38]. In the imaging context, this results in artifacts in the form of sharp peaks and valleys located randomly throughout the image field [1, 31]. Our group has taken two approaches towards the stabilization of these estimators. The first is based on kernel sieves as described in this proceedings [18]. The second is based on Good's [39, 40] roughness measure which we now describe. Denoting the density to be estimated in pixel i as λ_i, the 1-dimensional roughness is given by

$$\sum_i \frac{|\lambda_{i+1} - \lambda_i|^2}{\lambda_i} = \sum_i |\gamma_i - \gamma_{i-1}|^2, \tag{5.6}$$

with $\gamma_i = \sqrt{\lambda_i}$. For 2-dimensional applications in which the roughness is independent of position in the image, and for which there are no *natural* or *preferred* coordinate directions (see Roysam et al [4].) the rotationally invariant version is given by

$$\sum_i \sum_j [|\gamma_{i,j} - \gamma_{i-1,j}|^2 + |\gamma_{i,j} - \gamma_{i,j-1}|^2]. \tag{5.7}$$

The MRF determined by Good's rotationally invariant roughness becomes our model of the intensity within each object.

Because of the complexity of the more realistic tomography problem (both the requirement of large amounts of program memory as well as reconstruction computation time due to the large kernels), we have not yet implemented the attribute layer of the hierarchy, but have instead formed the edge sites using the Marr-Hildreth operator. As we have emphasized in the previous section, for smooth objects at high signal-to-noise ratios, the edge site placement using the edge operator is similar in quality and performance to the attribute approach. We are presently investigating more complex textured lung and brain image reconstructions for which the attribute layer is extremely important.

The constrained estimation problem with the controlling edge site layer becomes the following. Assuming the Poisson distribution of the measurements, then the problem becomes maximize with respect to $\{\lambda, e\}$ the following log-posterior distribution:

$$L(\lambda) - \alpha_1 \sum_i \left[\frac{|\lambda_{i+1} - \lambda_i|^2}{\lambda_i}\right] (1 - e_i) \tag{5.8}$$

with $L(\lambda)$ given in (5.5). The 2-dimensional version is straightforwardly generated using the rotationally invariant roughness of (5.7), and the introduction of horizontal and vertical edge sites.

Shown in Figure 3 is the heart phantom used for our tomographic reconstructions. For the simulations shown in Figure 4 a total of 16 view angles of data were collected, with point-spreads chosen to correspond to the Super PETT-I parameters. Shown in Figure 3 are the reconstructions of the heart phantom from $100K$ (top row) and $300K$ (bottom row) total simulated counts. The left column shows the result of maximizing the log-likelihood $L(\lambda)$ of (5.5), without the addition of Good's roughness measure. Notice the extremely rough structure of the unconstrained solution. Shown in the middle column is the result of adding Good's roughness measure, without any edge process. The right column shows the reconstruction resulting from the maximization of (5.8), while simultaneously estimating the pixel intensities as well as the edge process. The edge process was generated using the Marr-Hildreth operator at every iteration of the gradient descent on (5.8); the edge process is shown superimposed over the reconstruction.

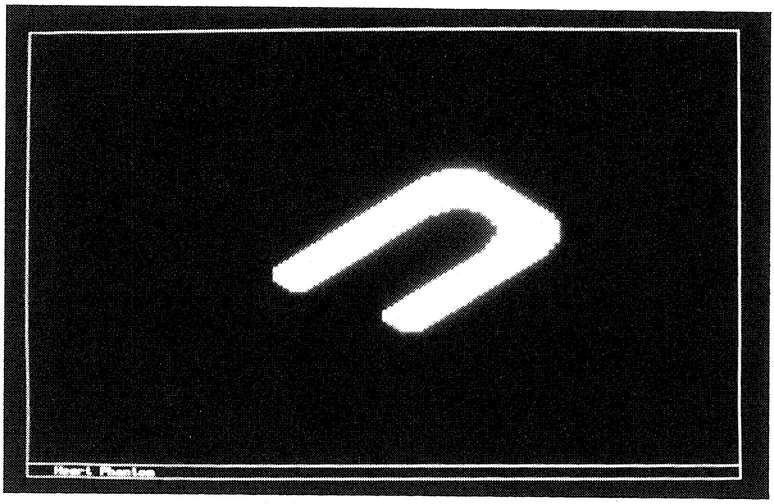

Figure 3: Original heart phantom for the PET reconstruction.

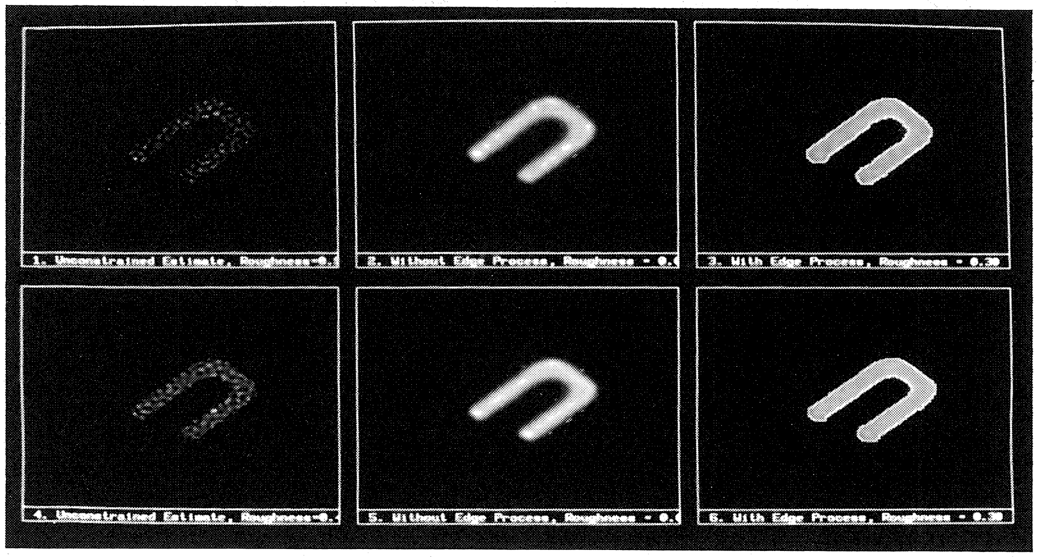
1. Unconstrained Estimate, Roughness=0.
2. Without Edge Process, Roughness = 0.
3. With Edge Process, Roughness = 0.30
4. Unconstrained Estimate, Roughness=0.
5. Without Edge Process, Roughness = 0.
6. With Edge Process, Roughness = 0.30

Figure 4: The top left shows the unconstrained EM algorithm from 16 view angles based on the Super PETT-I parameters with a total of 100 K counts. The top middle panel shows the reconstruction with Good's roughness applied, without an edge process. The top right panel the reconstruction with Good's roughness and the edge site layer active. The bottom row is identical but for 300 K counts.

6. Conclusions

We have proposed a method for mapping regular grammars to their unique Gibbs' representations, thereby allowing for a unified solution of stochastic and grammar based estimation problems. Coupling these results to the stochastic diffusions algorithms for sampling these Gibbs distributions yields a highly parallel method for sampling these rule-constrained sets. We present results computed on the DAP 510 parallel processor of Active Memory Technology for image segmentation and image reconstruction in positron emission tomography.

7. Acknowledgements

We wish to acknowledge support from the NSF via PYIA number ECE-8552518 and from the NIH grant RR01380.

References

1. Miller, M. I., & Snyder, D. L. (1987). The Role of Likelihood and Entropy in Incomplete-Data Problems: Applications to Estimating Point-Process Intensities and Toeplitz Constrained Covariances. *Proceedings of the IEEE* **75** (3), 892–907.

2. Snyder, D. L., Miller, M. I., Thomas, L. J. Jr., & Politte, D. G. (1987). Noise and Edge Artifacts in Maximum-Likelihood Reconstruction for Emission Tomography. *IEEE Transactions on Medical Imaging* **MI-6** (3), 228–237.

3. Good, I. J., & Gaskins, R. A. (1971). Nonparametric roughness penalties for probability densities. *Biometrika* **58** (2), 255–277.

4. Roysam, B., Shrauner, J. A., & Miller, M. I. (1988). *Bayesian imaging using good's roughness measure-implementation on a massively parallel processor.* ICASSP, 88.

5. Miller, M. I., Larson, K. B., Saffitz, J. E., Snyder, D. L., & Thomas, L. J. Jr. (1985). Maximum-likelihood estimation applied to electron-microscope autoradiography. *Journal of Electron Microscopy Techniques* **2**, 611-636.

6. Hopfield, J. J. (1982). Neural networks and physical systems with emergent collective computational abilities. *Proceedings National Academy of Science* **79**. 2554–2558.

7. Hopfield, J. J., & Tank, D. W. (1985). Neural Computations of Decisions in Optimization Problems. *Biological Cybernetics* **52**, 141–152.

8. Geman, S., & Geman, D. (1984). Stochastic Relaxation, Gibbs Distributions, and the Bayesian restoration of Images. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **PAMI–6** (6), 721–741.

9. Stallings, W. (1985). *Data and Computer Communications.* Macmillan Publishing Company, New York.

10. Chomsky, N. (1956). Three models for the description of language. *IRE Transactions on Inform. Theory* **IT–2**, 113-124.

11. Chomsky, N., & Miller, G. A. (1958). Finite State Languages. *Information and Control* **1**, 91–112.

12. Chomsky, N. (1959). On Certain Formal Properties of Grammars. *Information and Control* **2**, 147–167.

13. Grenander, U. (1976). *Pattern Synthesis: Lectures in Pattern Theory, I,* Springer-Verlag, New York.

14. Shannon, C. E., & Weaver, W. (1949). *The Mathematical Theory of Communication.* University of Illinois Press.

15. Gallager, R. G. (1968). *Information Theory and Reliable Communication.* John Wiley and Sons, Inc.

16. Jaynes, E. T. (1957). Information theory and statistical mechanics. *Phys. Rev.* **106**, 620-630.

17. Jaynes, E. T. (1983). *E. T. Jaynes: Papers on probability statistics and statistical physics.* D. Reidel Publishing Company.

18. Snyder, D. L., Politte, D. G., & Miller, M. L. (1988). *A Case Study in Statistical Image Processing: Positron-Emission Tomography,* AMS-IMS-SIAM Joint Conference on Spatial Statistics and Imaging, American Mathematical Society.

19. Karlin, S., & Taylor, H. M. (1981). *A Second Course in Stochastic Processes.* Academic Press, Inc.

20. Geman, S., Hwang, C.-R. (1987). *Diffusions for Global Optimization.* SIAM J. Control and Optimization.

21. Aluffi-Pentini, F., Parisi, V., & Zirilli, F. (1985). Global Optimization and Stochastic Differential Equations. *Journal of Optimization Theory and Applications* **47** (1).

22. Vichniac, G. Y. (1984). Simulating Physics with Cellular Automata. *Physica* **10D**, 96–116.

23. Miller, M. I., Roysam, B., & Smith, K. R. (1988). Mapping Rule-based and Stochastic Constraints to Connection Architectures: Implication for Hierarchical Image Processing. *SPIE Visual Communications and Image Processing, '88,* **1001**, 1078–1084.

24. Baker, J. K. (1975). The DRAGON System–An Overview. *IEEE Transactions on Acoustics, Speech, and Signal Processing* **ASSP-23** (1), 24–29.

25. Tsai, W. H., & Fu, K. S. (1980). A Syntactic-Statistical Approach to Recognition of Industrial Objects. *Proceedings of the 5th International Conference on Pattern Recognition*, 251–259.

26. Fu, K. S. (1986). A Step Towards Unification of Syntactic and Statistical Pattern Recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **PAMI–8** (3), 398-404.

27. Georgeff, M. P. (1982). Procedural Control in Production Systems. *Artificial Intelligence* **18**, 175–201.

28. Salomaa, A. (1977). *Formal Languages.* Academic Press.

29. Snyder, D. L., Thomas, L. J. Jr., & Ter-Pogossian, M. M. (1981). A mathematical model for positron emission tomography systems having time-of-flight measurements. *IEEE Transactions on Nuclear Science* **NS-28**, 3575–3583.

30. Miller, M. I., Snyder, D. L., & Miller, T. (1985). Maximum likelihood reconstruction for single photon emission computed tomography. *IEEE Transactions on Nuclear Science* **NS-32** (1), 769–778

31. Snyder, D. L., & Miller, M. L. (1985). The use of sieves to stabilize images produced with the EM algorithms for emission tomography. *IEEE Transactions on Nuclear Science* **NS-32**.

32. Nahi, N. E., & Jahanshahi, M. H. (1977). Image boundary estimation. *IEEE Trans. Computers* **C-26**, 772–781.

33. Derin, H., & Elliott, H. (1987). Modeling and Segmentation of Noisy and Textured Images Using Gibbs Random Fields. *IEEE Transaction on Pattern Analysis and Machine Intelligence* **PAMI–9** (1), 39–55.

34. Geman, D., Geman, S., Graffigne, C., & Dong, P. (1988). *Boundary detection by constrained optimization.* preprint.

35. Martelli, A. (1976). An application of heuristic search methods to edge and contour detection. *Comm. ACM* **19**, 73–83.

36. Marr, D., & Hildreth, E. C. (1980). Theory of edge detection. *Proc. R. Soc. Lond.* **B** (207), 187–217.

37. Grenander, U. (1981). *Abstract Inference.* John Wiley and Sons.

38. De Montricher, G. F., Tapia, R. A., & Thompson, J. R. (1975). Nonparametric Maximum Likelihood Estimation of Probability Densities by Penalty Function Methods. *The Annals of Statistics* **3** (6), 1329–1348.

39. Good, I. J. (1971). A Non-Parametric Roughness Penalty for Probability Densities. *Nature, London* **229**, 29–30.

40. Good, I. J. (1981). Roughness Penalties, Invariant Under Rotation for Multidimensional Probability Density Estimation. *J. Statist. Comput. Simul.* **12, 13**, 142–144.