LA-UR- 04-2587

Approved for public release; distribution is unlimited.

Title: AN AGENT BASED MODEL OF GENOTYPE EDITING

Author(s): Luis M. Rocha,
Chien-Feng Huang,

Submitted to: 8th International Conference on
Parallel Problem Solving from Nature (PPSN VIII):

An Agent Based Model of Genotype Editing

Luis M. Rocha[1] and Chien-Feng Huang[1]

Modeling, Algorithms, and Informatics Group (CCS-3),
Computer and Computational Sciences,
Los Alamos National Laboratory, MS B256,
Los Alamos, NM 87545, USA
{rocha, cfhuang}@lanl.gov

Abstract. This paper presents our investigation on an agent-based model of Genotype Editing. This model is based on several characteristics that are gleaned from the RNA editing system as observed in several organisms. The incorporation of editing mechanisms in an evolutionary agent-based model provides a means for evolving agents with heterogenous post-transcriptional processes. The study of this agent-based genotype-editing model has shed some light into the evolutionary implications of RNA editing as well as established an advantageous evolutionary computation algorithm for machine learning. We expect that our proposed model may both facilitate determining the evolutionary role of RNA editing in biology, and advance the current state of research in agent-based optimization.

1 Introduction

Evidence for the important role of non-protein coding RNA (ncRNA) in complex organisms (higher eukaryotes) has accumulated in recent years. "ncRNA dominates the genomic output of the higher organisms and has been shown to control chromosome architecture, mRNA turnover and the developmental timing of protein expression, and may also regulate transcription and alternative splicing." ([10], p 930).

RNA Editing ([2]; [1]), a process of post-transcriptional alteration of genetic information, can be performed by ncRNA structures (though it can also be performed by proteins). The term initially referred to the insertion or deletion of particular bases (e.g. uridine), or some sort of base conversion. Basically, RNA Editing instantiates a non-inheritable stochastic alteration of genes, which is typically developmentally and/or environmentally regulated to produce appropriate phenotypical responses at different stages of development or to states of the environment.

The most famous RNA editing system is that of the African Trypanosomes [2]. Its genetic material was found to possess strange sequence features such as genes without translational initiation and termination codons, frame shifted genes, etc. Furthermore, observation of mRNA's showed that many of them were significantly different from the genetic material from which they had been transcribed. These facts suggested that mRNA's were edited post-transcriptionally.

It was later recognized that this editing was performed by guide RNA's (gRNA's) coded mostly by what was previously thought of as non-functional genetic material [16]. In this particular genetic system, gRNA's operate by inserting, and sometimes deleting, uridines. To appreciate the effect of this edition let us consider Fig. 1. The first example (p. 14 in [2]) shows a massive uridine insertion (lowercase u's); the amino acid sequence that would be obtained prior to any edition is shown on top of the base sequence, and the amino acid sequence obtained after edition is shown in the gray box under the base sequence. The second example shows how, potentially, the insertion of a single uridine can change dramatically the amino acid sequence obtained; in this case, a termination codon is introduced. It is important to retain that a mRNA molecule can be more or less edited according to the concentrations of the editing operators it encounters. Thus, several different proteins coded by the same gene may coexist in an organism or even a cell, if all (or some) of the mRNA's obtained from the same gene, but edited differently, are meaningful to the translation mechanism.

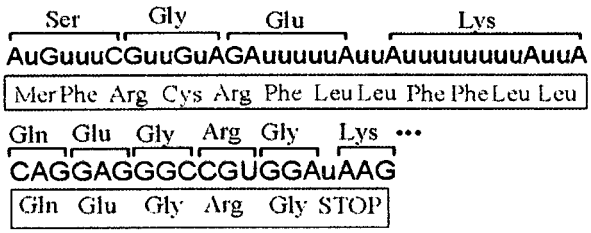

Fig. 1. U-insertion in Trypanosomes' RNA

The role of RNA editing in the development of more complex organisms has also been shown to be important. Lomeli et al. [9] discovered that the extent of RNA editing affecting a type of receptor channels responsible for the mediation of excitatory postsynaptic currents in the central nervous system, increases in rat brain development. As a consequence, the kinetic aspects of these channels differ according to the time of their creation in the brain's developmental process. Another example is that the development of rats without a gene (ADAR1) known to be involved in RNA editing, terminates midterm [17]. This showed that RNA Editing is more prevalent and important than previously thought. More recently, Hoopengardner et al. [6] found that RNA editing plays a central role in nervous system function. Indeed, many edited sites recode conserved and functionally important amino acids, some of which may play a role in nervous system disorders such as epilepsy and Parkinson Disease.

Although RNA editing seems to play an essential role in the development of some genetic systems and more and more editing mechanisms have been identified, not much has been advanced to understand the potential evolutionary advantages, if any, that RNA editing processes may have provided. To acquire insights for answering this question, we started a systematic study of a Genetic Algorithm with Edition (GAE) initially proposed by Rocha [13], [14]. Specif-

ically, we have employed a simple GAE model and reported some results on how Genotype Editing may provide evolutionary advantages ([7], [8] and [15]). Here, we continue this study by presenting further results obtained from a more realistic, agent-based model of Genotype Editing. Our goal is to gain a deeper understanding of the nature of RNA editing and exploit its insights to improve evolutionary computation tools and their applications to complex problems. In the next section, we summarize our prior work in Genetic Algorithms with Genotype Edition and discuss how we build on this work to produce the agent-based model for Genotype Edition.

2 Modeling Genotype Edition

2.1 Genetic Algorithm with Edition

In science and technology Genetic Algorithms (GA) [5] have been used as computational models of natural evolutionary systems and as adaptive algorithms for solving optimization problems. Table 1 depicts the process of a simple genetic algorithm.

Table 1. Mechanism of a simple GA.

1. Randomly generate an initial population of l agents
each defined by a n-bit genotype string (a.k.a. chromosome).
2. Evaluate each agent's (phenotype) fitness.
3. Repeat until l offspring have been created.
a. select a pair of parent agents for mating;
b. apply genotype crossover operator;
c. apply genotype mutation operator.
4. Replace the current population with the new population.
5. Go to Step 2 until terminating condition.

GAs operate on a population of artificial organisms, or agents. Each agent is comprised of a genotype and a phenotype. Evolution occurs by iterated stochastic variation of genotypes, and selection of the best phenotypes in an environment according to a fitness function. In machine learning, the phenotype is a candidate solution to some optimization problem, while the genotype is an encoding of that solution by means of a domain independent representation, namely, binary strings (or chromosomes). In traditional GAs, this code between genotype and phenotype is a direct and unique mapping. In biology, however, there exists a multitude of processes, taking place between the transcription of genes and their expression, responsible for the establishment of a one-to-many relation between genotype and phenotype. For instance, it was shown that RNA editing has the power to dramatically alter gene expression [12] (p. 78): "cells with different mixes of (editing mechanisms) may edit a transcript from the same gene differently, thereby making different proteins from the same opened gene."

In a genetic system with RNA editing, in other words, before a gene is translated into the space of proteins it may be altered through interactions with other

types of molecules, namely RNA editors such as gRNA's. Based upon this analogy, Rocha [13], [14] expanded the traditional GA with a process of stochastic edition of the genotypes of agents, prior to being translated into phenotypes (solutions). The editing process is implemented by a set of editors with different editing functions, such as insertion or deletion of symbols in the original genotype strings. Before these genotype strings can be translated into the space of phenotypes, they must "pass" through successive layers of editors, present in different concentrations. In each generation, each genotype string has a certain probability (given by the concentrations) of encountering an editor in its layer. If an editor matches some subsequence of the genotype string when they encounter each other, the editor's function is applied and the genotype string is altered. The GA with Edition (GAE), defined in [7], [8] and [15], is summarized in the following paragraphs:

The GAE model consists of a family of r m-bit strings, denoted as $(E_1, E_2, \ldots, E_r)$, that is used as the set of editors for the genotypes of the agents in a population. The length of the editor strings is assumed much smaller than that of the genotype strings: $m << n$, usually an order of magnitude. An editor E_j is said to match a substring, of size m, of a genotype string, S, at position k if $e_i = s_{k+i}, i = 1, 2, \ldots, m$, $1 \leq k \leq n - m$, where e_i and s_i denote the i^{th} bit value of E_j and S, respectively. For each editor, E_j, there exists an associated editing function, F_j, that specifies how a particular editor edits the genotype strings. For instance, when the editor matches a portion of a genotype string, a number of bits are inserted into or deleted from the latter.

If the editing function of editor E_j is to add one specific allele at s_{k+m+1} when E_j matches S at position k, then all alleles of S from position $k+m+1$ to $n-1$ are shifted one position to the right (the allele at position n is removed). Analogously, if the editing function of editor E_j is to delete an allele, an allele at s_{k+m+1} is deleted when E_j matches S at position k. All the alleles after position $k+m+1$ are shifted in the inverse direction (one randomly generated allele is assigned at position n).

Finally, let the concentration of the editor family be defined by $(v_1, v_2, \ldots, v_r)$, where the concentration of editor E_j is denoted as v_j: the probability that S encounters E_j. Figure 2 depicts the model. With these settings, the algorithm for the GAE is essentially the same as the regular GA, except that step 2 in Table 1 is now redefined as:

"For each agent's genotype in the population, apply each editor E_j with probability v_j (i.e., concentration). If E_j matches the agent's genotype string S, then edit S with the editing function associated with E_j and evaluate the resulting agent's fitness."

It is important to notice that the "post-transcriptional" edition of genotypes is not a process akin to mutation, because editions are not inheritable. Just like in biological systems, it is the unedited genotype that is reproduced. One can also note that Genotype Editing is not a process akin to the Baldwin effect as we discussed in previous work ([8], [15]).

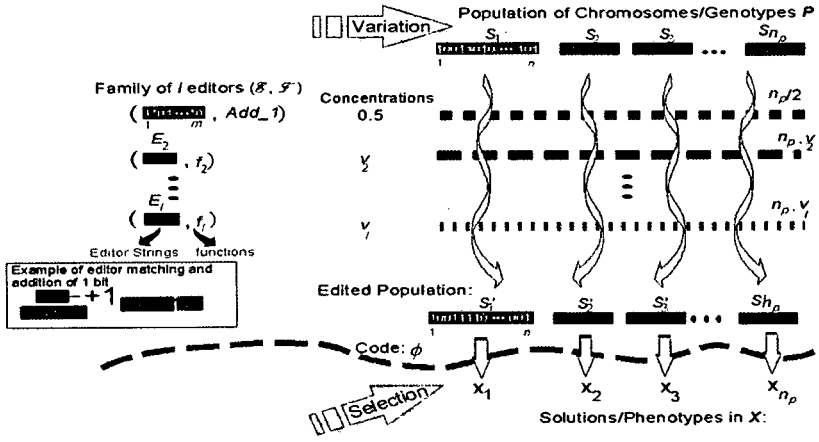

Fig. 2. Schematic of GAE

2.2 Agent Based Model of Genotype Editing

In this paper, we extend the simple GAE model to a more realistic agent-based GAE model. Whereas the GAE model defines a single family of editors for the entire population, the agent-based model we introduce here allows for heterogeneous agents, each with a distinct editor family. Therefore, instead of every chromosome encountering the same editors with the same probability, in the agent-based model of Genotype Editing (ABMGE) each agent's chromosomes are edited by its own editor family. Figure 3 depicts an agent in the ABMGE model.

Table 2 shows the algorithm for the ABMGE. In this model, the editor family for each agent, once generated, is fixed, and crossover and mutation (step 3.b and 3.c) are applied only to the agents' genotypes.

One way to highlight the difference between the two models is to notice that in the GAE model, there are essentially two separate populations: an evolving population of genotype strings and a fixed, small, editor population (family). Moreover, in the simple GAE the entire population of agents faces exactly the same "post-transcriptional" editors. In contrast, in the ABMGE model, the evolving agents face heterogenous post-transcriptional editors. As fit agents are selected for reproduction, their editor families propagate to the next generation. Here we do not allow for variation of the editor family; thus, there is no evolution of better editors. We leave such a study for a future article, and focus here exclusively on allowing heterogeneous genotype edition in an agent population.

In static and dynamic environments our results in [7], [8] and [15] have demonstrated how homogeneous genotype editing, as implemented in the GAE model,

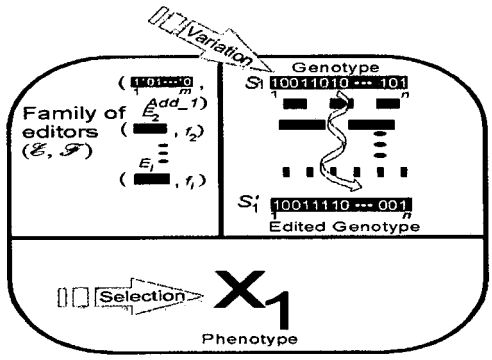

Fig. 3. Schematic of an agent in ABMGE

Table 2. Mechanism of ABMGE.

1. Randomly generate an initial agent population, each agent consisting of a n-bit chromosome and a family of editors.
2. Edit each agent's chromosome using the agent's editor family and evaluate each agent's fitness.
3. Repeat until l offspring have been created.
a. select a pair of parents for mating;
b. apply genotype crossover operator;
c. apply genotype mutation operator.
4. Replace the current population with the new population.
5. Go to Step 2 until terminating condition.

can improve the standard GA search performance by suppressing the effects of hitchhiking. We have also showed that editing frequency plays a critical role in the evolutionary advantage provided by the editors: only a moderate degree of editing processes facilitates the exploration of the search space. Therefore, one needs to choose proper editor parameters to avoid over or under-editions. We offered guidelines for choosing editing parameters in those publications. Here, we present a comparative study of GAE and ABMGE, and demonstrate how this agent-based model can enhance the search performance.

3 Empirical Results

How rapid is evolutionary change, and what determines the rates, patterns, and causes of change, or lack thereof? Answers to these questions can tell us much about the evolutionary process. The study of evolutionary rate in the context of GA usually involves defining a performance measure that embodies the idea of rate of improvement, so that its change over time can be monitored for investigation. In many practical problems, a traditional performance measure is the "best-so-far" curve that plots the fitness of the best individual that has

been seen thus far by generation n. As a step towards a deeper understanding of how Genotype Editing works, we employ a testbed, the small "Royal Road" S_1 due to its simplicity for tracing the evolutionary advancement [7].

Table 3. Small royal road function S_1

s_1 = 11111**********************************	c_1 = 10
s_2 = *****11111*****************************	c_2 = 10
s_3 = **********11111*************************	c_3 = 10
s_4 = ***************11111********************	c_4 = 10
s_5 = ********************11111***************	c_5 = 10
s_6 = ************************11111**********	c_6 = 10
s_7 = ****************************11111*****	c_7 = 10
s_8 = **********************************11111	c_8 = 10

Table 3 illustrates the schematic of the small Royal Road function S_1. This function involves a set of schemata $S = \{s_1, \ldots, s_8\}$ and the fitness of a genotype bit string (chromosome) x is defined as $F(x) = \sum_{s \in S} c_s \sigma_s(x)$, where each c_s is a value assigned to the schema s as defined in the table; $\sigma_s(x)$ is defined as 1 if x is an instance of s and 0 otherwise. In this function, the fitness of the global optimum string (40 1's) is $10 \times 8 = 80$.

We have shown that several factors play a role in the GAE's search power – e.g., *size of the family of editors*, *editor length*, *editor concentration* and *editor function* [7], [8]. Since a multitude of parameter combinations are possible, we conduct numerous ABMGE runs where the four parameters above are randomly generated in the beginning of each run and then fixed until the end of that run. The results are then averaged over the number of the runs so that we may compare the performance discrepancy of different search algorithms.

The settings of the editor parameters in each ABMGE run are: the size of editor family, r, is a randomly generated integer from 1 to 5; each editor is a randomized bit-string of a randomly chosen number of bits from 1 to 5 (which is fixed at the beginning of each run); the editor concentration is randomly generated in [0,1]; and the editor function inserts or deletes a randomly chosen number of bits from {1,2,3}, as well.[1] In addition, throughout this section, we always use a population of 40 agents where each agent is comprised of a genotype string, selection is via binary tournament [3], and crossover and mutation rates of 0.7 and 0.005, respectively.

Figure 4 displays the results on averaged best-so-far performance over 300 runs for the ABMGE, GAE and traditional GA.[2] One can see that the ABMGE clearly outperforms the GAE, and also outperforms the traditional GA (with the same parameters as the ABMGE, but without editors). In the ABMGE, editor

[1] In the case of insertion, the editor adds a random substring each time, but the length of the substring (the number of bits) is fixed throughout the course of each run.

[2] The value of the averaged best-so-far performance is calculated by averaging the best-so-fars obtained at each generation for all 300 runs; and so is the averaged editing frequency, where the vertical bars overlaying the performance measure curves represent the 95-percent confidence intervals. This applies to all the results presented in this paper.

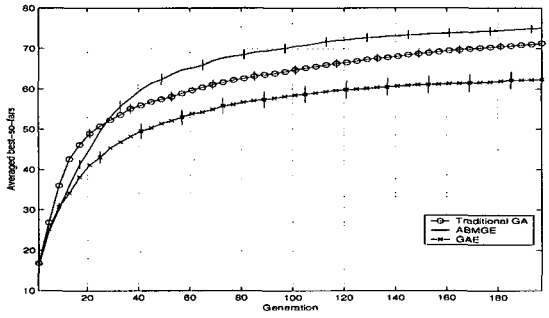

Fig. 4. Averaged best-so-far performance

families that generate fit agents can propagate with the genetic information of agents, thus providing an evolutionary advantage. These results highlight that the ABMGE, by allowing heterogeneous editor families, is thus evolutionarily advantageous.

Figure 5.a depicts the averaged editing frequency (the total number of times all editors edited chromosomes in a generation) for the ABMGE, which is substantially smaller than that of the GAE. One can notice that in the case of the GAE, where the averaged best-so-fars attained is far from the optimum, the editing frequency does not significantly drop to zero near the end of the experiments. It appears that the GAE's agent population continues utilizing the editors to explore the search space. Indeed, its corresponding population diversity, displayed in Figure 5.b, is far from zero.[3] This indicates that the system settles into a dynamic equilibrium in which the exploratory power of the editing process is balanced by the exploitative pressure of selection.

In the case of the ABMGE, whose best-so-far fitness is much closer to the optimum, the striking difference is that the corresponding editing frequency declines dramatically as the ABMGE's population evolves, and tends to drop significantly at the end of the experiments. This shows that the editing process, when advantageous editing occurs, ultimately comes to almost an end and the population diversity is considerably lost. These results are consistent with (but better than) what we have obtained in [7] and [8] for homogeneous genotype edition.

[3] To measure diversity at the i^{th} locus of a GA string, a simple bitwise diversity metric is defined as [11]: $D_i = 1 - 2|0.5 - p_i|$, where p_i is the proportion of 1s at locus i in the current generation. Averaging the bitwise diversity metric over all loci offers a combined allelic diversity measure for the population: $D = \frac{\sum_{i=1}^{l} D_i}{l}$. D has a value of 1 when the proportion of 1s at each locus is 0.5 and 0 when all of the loci are fixed to either 0 or 1. Effectively it measures how close the allele frequency is to a random population (1 being closest).

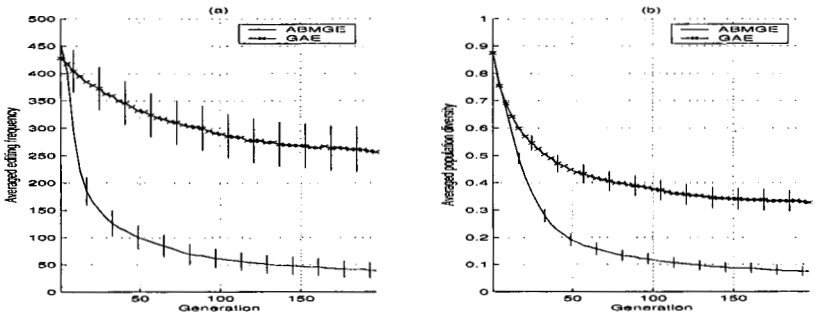

Fig. 5. Averaged editing frequency and population diversity

4 Conclusion and Future Work

We have presented a comparative study of ABMGE and GAE based on four editor parameters – size of the family of editors, editor length, editor concentration and editor function. We have demonstrated that the agent-based model of heterogeneous Genotype Edition can improve the search performance of traditional GA and GAE.

We have also shown that as the population of agents converges to a single phenotype (or a few phenotypes), editing frequency typically dramatically decrease so that the editing process ultimately comes to an end. It is interesting to note that this insight is consistent with phenomena observed in biology. Indeed, we know [2] that in the course of evolution RNA Editing was partially or completely eliminated in many lineages of eukaryotic organisms containing mitochondria, by reverse transcription of partially edited mRNA's, which validates our simulation results above. In this sense, our results share some superficial similarities with the work of Hinton and Nowlan [4], but we have discussed the differences elsewhere ([8] and [15]).

In this paper we have thus far discussed the ABMGE solely with constant parameters (in each run), such as fixed concentrations, of editors and a stable environment. In future work we will allow variation to be applied to editor families, thus enabling proper co-evolution of editors and genotypes. Furthermore, in order to investigate how RNA Editing may be advantageous in dynamic environments, we will also allow the concentrations of editors to be associated with environmental changes in order to introduce a control mechanism leading to phenotypic plasticity and greater evolvability. We have started such a study for the simpler GAE with good results [15], but intend to extend it to the ABMGE here presented. Together with the insights acquired previously, we expect that this research will enable us to (1) conduct more biologically realistic experiments which may lead us towards a better understanding of the advantages of RNA editing in nature, and (2) develop novel agent-based computation tools for dealing with complex, dynamic real-world problems.

References

1. Bass, B.L. (Ed.) (2001). RNA Editing. Frontiers in Molecular Biology Series. Oxford University Press.
2. Benne, R. (Ed.) (1993). RNA Editing: The Alteration of Protein Coding Sequences of RNA. Ellis Horwood.
3. Goldberg, D. E. and Deb, K. (1991). "A Comparative Analysis Of Selection Schemes Used In Genetic Algorithms." Foundation of Genetic Algorithms, Morgan Kaufmann, pp. 69-93.
4. Hinton, G. E. and Nowlan, S. J. (1987). "How learning can guide evolution." Complex Systems. Vol. 1, pp. 495-502.
5. Holland, J. H. (1975). Adaptation in Natural and Artificial Systems. University of Michigan Press.
6. Hoopengardner, B., Bhalla, T., Staber, C., and Reenan, R. (2003). "Nervous System Targets of RNA Editing Identified by Comparative Genomics." Science 301: 832-836.
7. Huang, C-F. and Rocha, L. M. (2003). "Exploration of RNA Editing and Design of Robust Genetic Algorithms." Proceedings of the 2003 IEEE Congress on Evolutionary Computation, IEEE Press, pp. 2799-2806.
8. Huang, C-F. and Rocha, L. M. (2004). "A Systematic Study of Genetic Algorithms with Genotype Editing." Proceedings of the 2004 Genetic and Evolutionary Computation Conference (GECCO-2004), in press.
9. Lomeli, H. et al. (1994). "Control of Kinetic Properties of AMPA Receptor Channels by RNA Editing." Science, 266: 1709-1713.
10. Mattick, J. S. (2003). "Challenging the Dogma: the Hidden Layer of Non-protein-coding RNAs in Complex Organisms." BioEssays. 25: 930-939.
11. Mahfoud, S. W. (1995). Niching Methods For Genetic Algorithms. Ph. D. thesis, IlliGAL Report No. 95001. Urbana, IL: University of Illinois at Urbana-Champaign.
12. Pollack, R. (1994). Signs of Life: The Language and Meanings of DNA. Houghton Mifflin.
13. Rocha, Luis M. (1995). "Contextual Genetic Algorithms: Evolving Developmental Rules." Advances in Artificial Life. Springer Verlag, pp. 368-382.
14. Rocha, Luis M. (1997). Evidence Sets and Contextual Genetic Algorithms: Exploring Uncertainty, Context and Embodiment in Cognitive and biological Systems. PhD. Dissertation. State University of New York at Binghamton. Science.
15. Rocha, L. M. and Huang, C.-F. (2004). "The Role of RNA Editing in Dynamic Environments." The Ninth International Conference on the Simulation and Synthesis of Living Systems (ALIFE9), in press.
16. Sturn, N. R. and Simpson, L. (1990). "Kinetoplast Dna Minicircles Encode Guide Rna'S for Editing of Cytochrome Oxidase Subunit Iii Mrna." Cell, 61: 879-884.
17. Wang, Q., Khillan, J., Gadue, P., and Nishikura, K. (2000). "Requirement of the RNA Editing Deaminase Adar1 Gene for Embryonic Erythropoiesis." Science, 290 (5497): 1765-1768.
18. Yamanaka, S. et al. (1997). "A Novel Translational Repressor Mrna is Edited Extensively in Livers Containing Tumors Caused by The Transgene Expression of the Apob Mrna-Editing Enzyme." Genes and Development, 11: 321-333.