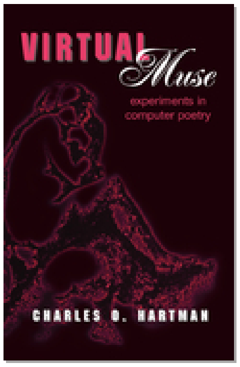
Virtual Muse

Charles O. Hartman

Published by Wesleyan University Press



>>>> / 8 /

AVENUES

"Seventy-Six Assertions and Sixty-Three Questions" was a new kind of poem for me, not only in its computer origin but in its style and sound. In its way, the computer was collaborating now. It was helping me think about poetry. Not simply confirming or codifying knowledge I already had, like the chorale-harmonization program, it was becoming a tool of discovery.

I kept exploring the possibilities of Prose. One of the simplest variations arose through an outside request. Every other year, Connecticut College's Center for Arts and Technology hosts a symposium, and at one of these I gave a brief talk about Prose. Afterward, Manfred Fischbeck, choreographer and teacher at the University of the Arts in Philadelphia, approached me to ask whether the dictionary that Prose used could be changed. I said yes, that I'd designed it that way on purpose. He told me about a new piece he was working on, involving dancers and film and music and words. We agreed on this: He would sit down with his dancers and get them to list words—nouns, verbs, and adjectives—associated with dance and with this project. I would make a new dictionary, keeping all the necessary "linguistic glue" words like conjunctions and pronouns but replacing the usual long lists of referential words with the relatively short lists of words the dancers chose. Then I would generate text and edit from it about two minutes' worth of sentences to be spoken (by Manfred) as a voiceover during the first section of the dance. The result is printed in the appendix as "Dance Text." Here's the opening:

> The stage reverses a closed room, where every rehearsal draws
>   its unreal distance.

Repetition: the machine of memory.
Turbulence: a traveling repetition.
The reward of turbulence: balance.
Performance is language, but we think to feel.
To think is the beginning of work.
To imagine gives speed.
To fall is slowing down, and to accelerate is any jump.
Space becomes the page of dance, where we flow between the
  dream and the blue beat.
Deep time: so dark a figure.
Someone is a shape.
When we were these many gestures, you were these many colors.
While all the dancers are bending these rhythms, the cloud of
  hands calls the ballet across the face of the air.
To talk is a dimension; to organize is music.
The mechanics of dream connect these nerves in groups.

This was still editing normal Prose output. But I'd also been trying out different attitudes toward computer-generated text. One idea, always in the back of my mind, treated the computer's language as *oracular*. I was thinking partly of the Greek oracles—a word that means both the representatives of the gods to whom people took important questions and the puzzling truths with which they answered. The oracles usually turned out to be truthful in ways nobody understood until it was too late. Greeks from Oedipus through Euclid believed fervently in the truth but were fatalistic about its usefulness. Most human cultures seem to prize anonymous, cryptic statements embodying wisdom. Proverbs are another form of oracle. They can be very mysterious. Barbara Herrnstein Smith, a critic who has thought about proverbs as a special kind of literature, likes this German one: "When the wind blows, the tree shakes." There's almost nothing it can't refer to, no situation in which it *doesn't* offer prudent advice. It's up to us to interpret it correctly for each occasion. You could clear your throat at a tense moment in any meeting, utter that sentence, and sit back, confident in your reputation for insight.

This recalls the play between sense and nonsense we examined earlier. It may also bring us back to randomness. One kind of oracle used in the Middle Ages was the *sortes Virgilianae*; when you had a problem, you would open Virgil's *Aeneid* at random, point blindly to a line, and then read it as a commentary on your question. It's the same principle as in the I Ching. *Sortes* means "lots" (as in "casting lots") and finally comes to mean "fate" or "destiny". There's an old link between chance and necessity; both are names for what's outside our control.

Some cultures take their wisdom as it comes, some collect it in memory, and some even write it down. Then it becomes "scripture," which means "writing" but especially (while writing is a new and rare skill) holy writing, the transcribed word of a god or God. A few cultures go on to add layers of interpretation and commentary to this basic scripture. The Hindus are a good example. But no one has outdone the Jews in this respect. The clearest monument to this is the Talmud.

I'm no expert on the Talmud, but the outlines of its history are readily accessible. Moses brought down from Mount Sinai two laws: the Ten Commandments written on stone and a body of oral law that each generation of priests must teach to the next. This oral law, the *halakhah*, together with the oral commentary that grew up around it, became known as the Mishnah. During the five centuries between the destruction of the Temple and the destruction of the Roman Empire, the Mishnah was written down; and an enormous mass of further commentary, called the Gemara, began to be built on top of it. Relevant quotations from the Torah were added for confirmation and resolution of obscure points. Important commentaries by later rabbis also attached themselves to the Mishnah and Gemara. The whole thing became known as the Talmud, from a root word that means *learning*.

A page in a fine Talmud is visually striking. At the top of the page, in heavy Hebrew print, is a small rectangle of the Mishnaic text. Surrounding it on the bottom and both sides, in lighter type, is the long Gemara for that part of the Mishnah. There are many pages in each of many volumes. An old saying equates the Talmud with an ocean that a lifetime would barely suffice to swim across.

I began to think of imitating, from that enormous work, a principle of construction, a way to both invoke and modify people's normal poetry-reading reflexes. Taking output from Prose as my "Mishnah," my text for commentary, I could write something modeled on the Talmud. Of course the idea wasn't to comment on the Talmud itself. I wouldn't be likely to call my poem "A Talmud" or to quote from that masterwork of many authors. I wouldn't even treat similar topics. All I wanted was a design.

The great appeal of the Talmud's method is that it's so mutlivocal. There are voices upon voices upon voices: the *halakhah*; the prescriptural oral commentaries on it in the Mishnah; the anonymous original layers of the Gemara (collected by hundreds of teachers over hundreds of years); the further notes and clarifications and stories added by later, named rabbis; the references to the books of the Torah. Any section of the Talmud resounds with many personalities, no matter how intently they all focus on a single, minutely defined topic, such as the proper treatment of stones found underneath a tree that is worshipped by heathens.

Imitating this kind of communal work wouldn't be completely off the poetic track historically. The impulse toward multiple voices has informed a lot of modern poetry. The literary theorist Mikhail Bakhtin distinguished lyric poetry, which he called *monologic*, from the *dialogic* nature of novels. A novel echoes not just with the words of different characters but with the languages of different classes, professions, age groups, cliques, clans, and parties. Yet poetry, too, can be dialogic. In a sense, how could it not be? Our own selves aren't simple, single entities. As Bakhtin says, "The ideological becoming of a human being . . . is the process of selectively assimilating the words of others." Eliot's *Waste Land* and Pound's *Cantos* and Williams's *Paterson* in their various ways juxtaposed not only images but voices. This impulse has deepened in recent decades, whether in John Ashbery's strongly mixed diction or in Jerome Rothenberg's adaptations of tribal communions. For any poet who has grown dissatisfied with the monologic lyric voice, the Talmud might offer an intriguing model.

What could the voices of such a poem be? However disparate their origins, the computer's random prose, like the arcane Mishnah, seems to call for commentary. How would I go about supplying interpretation for it? Of course, I would rely on my own experience as a reader. But "I"—not only as a reader with an education and a history of my own but as a self created partly by dialogic interchange with the world around me—have potentially many voices.

For instance, what does a careful reader do when something isn't clear? Faced with the problem of interpreting an obscure text, one of my first impulses is to look up certain words in the dictionary. (How do I know which ones to look up?) The Torah is most often used in the Gemara for just the purpose—to establish the correct meaning of particular words. But the dictionary, too, is a kind of scripture. It's an anonymous systemization of all the essentials of human life. In argument or in Scrabble, we frequently appeal to its supreme authority. We say "the Dictionary"—though there are many dictionaries—much as we say "the Bible," which means "book." Furthermore, as a literary scholar, in a critical pinch I don't use just any dictionary. I turn to the *Oxford English Dictionary*, the fattest and most authoritative of all. The *OED*'s authority is itself dialogic, derived from the masses of quotations that bring into this one book the whole history of our literary culture.

Even without the dictionary, within my own private response to certain evocative sentences that emerge from the Prose machine, I can hear more than one character speaking. There's a finicky analyst of logic and grammar; a fellow who luxuriates in images conjured up (however remotely) by any fragment of phrase; a literary interpreter of figures of speech, and so on. I began to see a rare opportunity to work simultaneously as poet and as critic. The layers piled up before my eyes.

Getting the text to comment on would mean going back a step. Treating the Prose output as oracular involves treating it as quasi-sacred—it wouldn't be appropriate to *edit* it. So it was back to scouting through reams of random text, looking for nuggets of wisdom. As compared with the weary seeker of AutoPoet days, I would need only single sentences, not stretches where meter and syntax combine to make plausible poetry.

This experiment grew logically out of "Seventy-Six Assertions and Sixty-Three Questions," even though the model I was starting from was quite different. In both cases, the computer program offered text, and I offered interpretation. In the earlier poem, the interpretation was superimposed on the text as editing. In the newer experiment, the interpretation was added after the text and surrounded it without covering it up.

I'll give just one example: a sentence I found in a pile of Prose, and the commentary I built up around it:

> WHAT CANNOT THE GLASS OF AIR HURT? Glass clear as air lends air also frangibility. (The sparrow, deceived, died.) Thus shards; and the hurt at the heart of all flesh figures as laceration. And the same transparency that speaks of air breaking presents that glass which aids long vision, or close vision, the pain the pain that attends sight.[1] And the sand-glass—behold, "her glasse is runne," shards raining around naked feet are shards of time. Not "Whom" but "What"; for *time that bears all things digests all things, and not persons only.*[2] Is there a glint among these cinders? The word is "cannot," rather than the indicative "does not" or the minatory "will not"; does this suggest an option, the hurt nullified or evaded, the glass able but unwilling to wound? so that the poet has said, "Sweet, it hurts not"?
>
> [1] Likewise, the glass is a burning glass, vision a fire to the eye. And the looking glass, most treacherous of all.
>
> [2] And this word "What" echoes the sound of universal termination, which Babhli heard whispered on the breeze on the side of the mountain; beginning in aspiration (which the Anglo-Saxons rightly placed first in their orthographies, *hwat*), running through that brusque vocable/vowel, clapped shut between palate and tongue, the secret.

Trial passages like this one never came together into a finished work for me. But the method fed into a more recent long-term project, a kind of personal encyclopedia of one-page definitions of (the project says) everything under the sun. (It's called *Except to Be*, partly because its prose never uses any form of the verb *to be*.) The impulse behind that entirely noncomputer work evolved, at least in part, from my thinking about the computer-based experiment.

What's next? The obvious lack in Prose is semantics. At its cleverest, the program is still never talking *about* anything. I've suggested how language in poetry works in other important ways besides referring to or signifying things. But most poetry also refers to things in the normal way as well. To do that, a generator of language (person or program) has to know something about things as well as about syntax.

As a crude beginning I could add another set of tags to each word in the dictionary, indicating a set of "topics" to which the word is relevant. This seems to be the approach of Racter in *The Policeman's Beard Is Half-Constructed*, which I described in the Introduction. According to A. K. Dewdney, Racter works with sentence templates like this: "THE noun.an verb.3p.et THE noun.fd." "An," "et," and "fd" mean "animal, "eating," and "food," respectively. These identifiers are attached to selected words in the dictionary, so the program seeking a "noun.an" selects only from words with the ".an" tag. If the programmer builds in a tendency for the same tags to keep turning up in several sentences, the program will seem to stick to a topic.

Yet for general-purpose meaning making, this wouldn't be very promising. Who defines a "topic"? *Roget's Theasaurus* contains a "Synopsis of Categories" that places all the possible things to talk about in an orderly outline. That could be a starting point. But if you're stuck with Peter Roget's idea of how the world is divided up, it becomes difficult to say anything interesting—at least by chance.

What's required is a kind of road map of the semantic "space" through which we move when we're talking. But it's a "space" in far more dimensions than three, and sometimes it seems to change even while we're traversing it. Analyzing this and codifying it in a computer program is a job for legions of programmers. In fact, here we are in the true realm of artificial intelligence, where, for now at least, the experts will have to take over. Even they haven't yet had any overwhelming success, though they seem to be getting closer.

In any case, this is the Imitation Game again. The trap for poetry is that the more accurately the computer mimics human language, the more ordinary it becomes. In fact, the ordinariness is how we measure the accuracy of imitation. The perfect AI language machine would convince us (win the Turing game) by being rather dull, like a good secret agent. The computer poet wants a more unstable balance of the plain and the strange.

Partly because I was running up against a taller mountain than I had much interest in climbing, I began to think again about the other main approach to "computer poetry": not text generation but text manipulation. These are the two logical choices, which you could diagram as.

PROGRAM ———> TEXT

and

TEXT ———> PROGRAM———> TEXT

A program like Prose generates new text without any input; a program like Travesty transforms one text into another. The number of possible transformations must be huge, and I began looking for interesting ones.

One method came to my attention through a delightful book by Jackson Mac Low called *The Virginia Woolf Poems* (Burning Deck Press, 1986). In an endnote to the book, Jackson explained the "diastic" or "spelling-thru" technique he had used in writing the poems. The process began with a striking phrase from Virginia Woolf's *The Waves*: "ridiculous in Picadilly." He reread the novel, looking for the first word that, like "ridiculous," began with an r; then the next word following that had (like "ridiculous") *i* as its second letter; then the next whose third letter was *d*; and so on until he had "spelled through" the whole phrase. (There were other rules for line breaks, punctuation, and so on.) The resulting text would be made entirely out of Woolf's words but would have none of the usual English syntax. As I read the poems, I was startled by how evocative a text this arbitrary system could produce.

Jackson had done all his work by hand. I sat down and embodied his rules in a little program called Diastext. I sent it to him, and he has used it in the making of several books since.

I also sent it to Hugh Kenner, the author of Travesty. Probably at the same time, I sent him a computer-disk copy of a wonderful "text" I had found, a little pamphlet called *Sentences for Analysis and Parsing*, from the Thayer Street Grammar School in Providence, Rhode Island. (It's anonymous, and what may be the only copy sits in a modest folder in the Brown University library. I tracked down Samuel Stillman Greene, a mid-nineteenth-century Providence educator, as the probable author.) The little book consists of 457 sentences, ranging from "Dogs barked" and "Halt!" through grandiose entries like "He spoke in as noble accents as ever fell from human lips." I had already used the text in one peculiar, if noncomputer, way, writing a one-act play called *Beauties*, for four characters, *all* of whose dialogue is made up of sentences from the pamphlet.

It was Hugh's idea to run the schoolbook text first through his Travesty program a number of times and then to run those outputs through Diastext in turn. Using the same sort of diagram as before, we could see this as

TEXT ——> PROGRAM ——> TEXT ——> PROGRAM ——> TEXT

We sorted through a good many of the results (each of them a Travesty output followed by the diastic "spelling through" of that output), picked the ones we liked, and put titles on them. The result is a book called *Sentences*, which Sun & Moon (a California press largely devoted to the work of "language" poets) published in 1995. (It displays, as Hugh notes in the Afterword, "an odd fixation on cigars.") One section (too long to insert here) appears in the Appendix.

Jackson's "diastic" method of text manipulation is arbitrary, not random. The process is completely deterministic once the author has chosen the input text (*The Waves*) and the "seed" phrase ("ridiculous in Picadilly"). You could even eliminate the seed phrase by making the text itself its own seed. (With a big input text, this produces a really enormous output; but you could—arbitrarily—cut it off at some point.) Turning that into a program is very straightforward, because the method is already an algorithm.

This got me thinking about other arbitrary linguistic algorithms, and I remembered a very old one: the Cabalists had various systems for translating letters into numbers and determining the mystic significance of words by way of the numbers their letters added up to. Some of these systems were very complicated, but one is about the simplest imaginable: *A* = 1, B = 2, and so on. So the word *word* totals 60; *abracadabra* is 52.

If I didn't want to construct a system of mystic significances, what could I do with these numbers? I noticed that while every word has a unique total, each total corresponds to several or many words; how many depends roughly on how large the number is. Huge arrays of words cluster around values between about 20 and 100. Coincidences diminish above 100 and become sparser and sparser as the totals rise. This numerology, in other words, can be seen not as characterizing single words but as identifying *groups* of words. Well, poems—from a certain peculiar point of view—consist of groups of words.

I wrote a tiny program, Numerol, that would read an input text, then ask the user repeatedly for a number, and write out all the words in the text with that total. As a refinement, I had the program use the *modulus* of the number, so the program wrote out all words whose total was evenly divisible by user's number. In practice, this didn't make much difference; it just intensified the clustering of words around lower values. In any case I found the results most interesting when I gave the program numbers in the range between 100 and 150.

As input, I used a file containing the complete manuscript of a book of poems I was working on. This didn't just establish some more formal "authorship" for me in the output; in my own eyes at least, it put my unmistakable stamp on the results. Especially in the range of those higher totals, we're talking about words outside the realm of the inevitable: not *because* and *tree* and *that* but *improvised* and *Plymouth* and *fingerbowls*. Those three words all total 130, and each of them is a word with which I have strong personal associations. The groups of words the program offered me felt like a collection of mirrors.

These personal associations are—well, personal. A handful of three or four words might have private meaning but would make collective sense to someone else only by accident. On the other hand, a collection of these collections of words begins to build up a picture, visible even from the outside, of a person, at least as a bundle of obsessions and habits and linguistic quirks. I had chosen the words in writing the poems; the program gave me a way to examine those acts of choice and present them *as* acts of choice. After looking over dozens of groups, I selected the ones that felt more telling and treated each one as a section within a poem.

The words in each group have no syntactical relation to each other, of course. This suggested that I should present them as collections and not so much as *sequences* (like the words in a sentence or even a line). Poetry in this century has already developed techniques for treating words that way. They involve using the two-dimensional space on the page, undermining the one-dimensional sequence by which printed language usually imitates the stream of syntactical speech. So more of my own choice entered as I arranged the words on the page. Here's one section from the middle of the poem:

consequence

yourself,

everybody

thousands.   yourself.

shattering yourself

alternating twisting pressure

thousands

dissipates yourself

versions   tortured,

convenient

Finally, I selected one two-word group as my title, "Extraordinary Instruments." (Each word totals 172.) It's one of the few groups that happens to offer a coherent *phrase*; and the phrase resonated with my feeling, in looking through all these selections from a personal dictionary or code book I had hardly known I kept, of how powerfully single words identify a way of looking at the world. I wasn't sure whether the "instruments" were musical or surgical, but at this level of linguistic abstraction I wasn't even sure of the difference.

Later, a logical alternative to this method occurred to me. By using my own other poems as the source text, I had predetermined the diction of "Extraordinary Instruments," and part of the poem's effect was to explore that diction. (It's a pretty narcissistic poem, in an obscure way.) From an odd angle, I was secretly reviving the old idea of a "poetic diction"—a set of words proper for use in poetry. Rebellions against this idea, whether Wordsworth's or Pound's, have had two motives in various proportions: to expand the range of poetry and to democratize it. Why not, in the spirit of perversity, take this principle all the way and give *every* word an equal chance?

I hunted around for a text-only online dictionary of English and found one with over a hundred thousand words—barely ten or twenty per cent of English but a respectable vocabulary for a college graduate. Then I built a new filter program, much like Numerol, that would accept a number from the user and report back every word in the dictionary whose letters summed to that number. The lists were long, especially for numbers under 100. Instead of receiving from the program a little handful of words to arrange in a section, I could get a sizable vocabulary out of which to build a poem. The obvious rule was that, whatever number I chose, the resulting poem couldn't contain any words not in the list.

Running through my head for days, like an ad jingle, had been the phrase "That's glory," which I thought was Dylan Thomas or Samuel Beckett until I realized it was Humpty Dumpty. ("There's glory for you!" is how he gets into explaining that he pays words extra to mean what he wants.) The total for *that's* is 68; the apostrophe doesn't count. The list of 68-words, about a thousand of them, included some splendid collisions: *goofy* and *logos*, *diligence* and *swank* and so on. The challenge of building a coherent poem out of these—without the help of *the* (33) or *and* (19) or *a* (1)—wasn't one in which the computer could help me much. But working from the computer's list, I could know I wasn't letting a foreign word slip in:

> That's acceptable. That's goofy
> elating language: gleeful logos
> nobly jeering, lauding drily—
> that's doings. That's bagsful,
> that's unabated beauts. That's
> poems readably suave, trued,
> pleading diligence, calving jetsam,
> dangling acuter Damoclean dangers
> safely. That's swank. . . .

Naturally enough, the poem turned out to be about poems and the various kinds of talking they could do.

Once again, the advantage of the computer here lies in its perfect ignorance of language as such. The necessities of evolution vitally condition and limit our reception and emission of words. We need to survive in social situations where a tendency to hear (like one of Woody Allen's characters) "Did you?" as "Jew?" would at least complicate our lives. (We can afford to mishear songs, and we do; my friend Preston McClanahan thought it went, "For he's a jolly good fellow, / with so many candy knives.") Computers fail to acknowledge the probabilities of human talk, which gives the designers of speech-recognition machines nightmares. But by the same token, humans may find it hard to play as freely in the field of language as poetry invites.

How many random or arbitrary methods for manipulating or selecting language could there be? Infinitely many? And of those, how many are potentially interesting? If there are answers to these questions, we'll find hints of them only through a very large number of experiments. As a poet trained in the use of traditional forms—though I use them only intermittently—I saw a connection between the long history of exploration that has given us those forms and the kinds of experiments encouraged by the search for text manipulation.

A new form begins as someone's invention and then, maybe, proves useful to other poets. We know one Greek form as the "Sapphic stanza." It may have been worked out originally by her contemporary, Alcaeus, but it was a favorite of Sappho's, and she used it in writing some magnificent poems. We still give it her name, though for thousands of years poets from Catullus through James Merrill have been using it—and varying it. The differences between Greek and a language like English *require* some variations. (In the pattern that defines the form, we usually replace "quantity," or syllable duration, with stress.) It has a distinctive shape on the page, so any more or less regular stanza that more or less resembles it is likely to register as a variation on the Sapphic stanza. These variations in a form can work like mutations in biology, providing the material for evolution. Ezra Pound claimed that the sonnet began as someone's variation on the older canzone.

Will certain computer poetry methods catch on and establish themselves and evolve in similar ways? Or will we shift the demand for originality in poems toward meta-originality in method? Another decade or two should tell us.

I'll close with one more example. Talking about "deterministic" systems these days is likely to bring up the fashionable subject of chaos. "Chaos" is our name for the way a simple, deterministic system can turn out to have unpredictable results. The weather, for instance, is the result of physical laws—gravity, friction, thermal expansion—that every high school physics student knows. But a radio station that gives a thousand-dollar "guarantee" on its forecast for the day's high temperature, even when the guarantee is only for accuracy within five degrees either way, pays out a thousand dollars two or three times a month.

Many formulas are known by now that produce chaos when key variables approach some particular value. One of them, an "iterative function" that can be used to model population growth, looks like this:

$$r*p*(1-p) \longrightarrow p.$$

It becomes "chaotic" as r approaches 4. (It turns out that r can never reach 4; the equation explodes infinitely at that point.) Suppose we use this formula to select, over and over, from a little collection of twelve or fifteen worlds? We write the program, give it the little pile of words, and start plugging in values for r. At low values the results are infinitely repetitious and therefore boring—but sometimes *intrestingly* boring if we put them into the right context. The simplest context is a series, in which we can see the results growing more and more complex as r increases. As with increasing values of n in Travesty, something seems to be struggling toward coherence. It would be whimsical, but logical, to end the sequence with a section that uses the words to make perfectly good sense and call it "$r = 4.0$"—as if "perfectly good sense" were the result of chaos gone ballistic. One result is given in the Appendix, a poem called "And Finger Light Because Almost Finger Elsewhere." (The title phrase occurs in one of the results I didn't finally use.) A different group of words and different values for r would produce infinitely many different poems.