Sens public

The Digital Reception of *A Hundred Thousand Billion Poems*

Jonathan Baillehache

2021

Textures : l'objet livre du papier au numérique

URI: https://id.erudit.org/iderudit/1089666ar
DOI: https://doi.org/10.7202/1089666ar

See table of contents

Publisher(s)

Département des littératures de langue française

ISSN

2104-3272 (digital)

Explore this journal

Cite this article

Baillehache, J. (2021). The Digital Reception of *A Hundred Thousand Billion Poems*. *Sens public*, 1–13. https://doi.org/10.7202/1089666ar

Article abstract

Raymond Queneau's *Cent mille milliards de poèmes* (*CMMP*) was first intended as a poetry writing "machine" ("machine à fabriquer des poèmes"). Queneau used the word "machine" in his preface to designate a tool designed to help the reader compose his/her own sonnets. Immediately after its publication in 1961, poetry smitten computer scientists and poets interested in computer science digitalized Queneau's book. All computer portings could generate poems automatically, thus transforming *CMMP* into a proto-text generator. The digital reception of *CMMP* made a composition tool into a machine. This article investigates the possible meanings of the word "machine" in the context of the adaptation of Queneau's *CMMP* by D. Starynkevitch (1961) and Paul Braffort (1975).

Sens [public]

The Digital Reception of *A Hundred Thousand Billion Poems*

Jonathan Baillehache

Publié le 15-06-2021

http://sens-public.org/articles/1498

Abstract

Raymond Queneau's *Cent mille milliards de poèmes* (*CMMP*) was first intended as a poetry writing "machine" ("machine à fabriquer des poèmes"). Queneau used the word "machine" in his preface to designate a tool designed to help the reader compose his/her own sonnets. Immediately after its publication in 1961, poetry smitten computer scientists and poets interested in computer science digitalized Queneau's book. All computer portings could generate poems automatically, thus transforming *CMMP* into a proto-text generator. The digital reception of *CMMP* made a composition tool into a machine. This article investigates the possible meanings of the word "machine" in the context of the adaptation of Queneau's *CMMP* by D. Starynkevitch (1961) and Paul Braffort (1975).

Résumé

Cent mille milliards de poèmes (*CMMP*) de Raymond Queneau est conçu comme une machine à composer des poèmes. Queneau utilise le mot « machine » dans la préface des *CMMP* pour désigner un outil offert au lecteur désireux de fabriquer ses propres sonnets. Immédiatement après sa publication en 1961, des ingénieurs informatiques férus de poésie et des poètes intéressés par l'informatique portèrent le livre de Queneau sur ordinateur. Tous les portages informatiques des *CMMP* incluent la possibilité de générer des poèmes automatiquement, transformant ainsi les *CMMP* en un proto-générateur de texte. Cette réception numérique des *CMMP* a transformé un outil de composition en automate. Cet article étudie les sens possibles du mot « machine » dans le contexte de l'adaptation des CMMP par D. Starynkevitch (1961) et Paul Braffort (1975).

Mot-clés : machine, générateur de texte, portage informatique, *Cent mille milliards de poèmes*, Raymond Queneau

Keywords: machine, poetry generator, digital porting, *Cent mille milliards de poèmes*, Raymond Queneau

Contents

The Digital Reception of *A Hundred Thousand Billion Poems*

Jonathan Baillehache

CMMP and Exquisite Corpses

Queneau's original printed version of *Cent mille milliards de poèmes* (*CMMP*) was conceived as a "machine" that allows readers to potentially create a hundred thousand billion poems (almost all perfect sonnets) by combining 140 verses printed on movable strips of paper. Queneau described his book as a "*machine à fabriquer des poèmes*". Immediately after its publication in 1961, computer scientists interested in poetry and poets interested in computing began to port the book to computers. In this presentation, I investigate the different potential meanings of the word "machine" in the context of the adaptation of Queneau's books into computer programs.

The word "machine", when used by Queneau in the preface of *CMMP*, is to be read more as an instrument than an automaton. Queneau's instructional manual clearly encourages all readers to compose their own poems at will. By empowering the reader with a configurative function, Queneau aimed at fulfilling Lautréamont's program: "poetry must be made by all, not by one" (1961, II). Queneau insists in his preface that his book does not resemble the surrealists' "exquisite corpse" (*cadavre exquis*)[1], but instead is inspired by a

[1]In an exquisite corpse, different hands write successive fragments of a single text, without reading the full text that they have to continue writing, allowing for a great deal of randomness in the outcome.

combinatory children's book[2]. Like an "exquisite corpse", *CMMP* combines heterogeneous fragments of texts to produce a surprising text. However, the user of Queneau's machine is aware of all the fragments of text at every moment of his configuration.

Following Queneau's reasoning, *CMMP* would in fact be a branching poem, where the reader evolves sequentially from one node of verses to the next, in a linear fashion, much like in a "Chose Your Own Adventure" book. In other words, Queneau was making clear that *CMMP* was not automatic writing. However, all computer adaptations of *CMMP* have integrated the possibility to use a randomizer that generates poems automatically, transforming *CMMP* into a proto-text generator. Randomness is a dominant feature in all the digital versions of *CMMP* whether they are composed by Oulipo members or not. The digital reception of *CMMP* seems to have transformed it from an instrument into a series of automata. For good reasons, scholars of electronic literature often hesitate between classifying *CMMP* as an ancestor of generative poetry or as an ancestor of hypertext. I'd like to suggest that this transformation from instrument to automaton is not a misreading of *CMMP*, but reveals in fact something latent in Queneau's original book.

Turing's Epigraph

The 1961 edition of *CMMP* cites as an epigraph a quotation from Alan Turing stating that "only a machine can appreciate a sonnet written by another machine". This quotation is reminiscent of Turing's debate on artificial intelligence. In his famous 1950's article "Computing Machinery and Intelligence" (1950), Turing proposes to address the question, "can computers think?" with an imitation game. In the game, a human converses with another human and with a computer using written messages, and must guess which one is which based on the answers he receives. Turing demonstrates that there will always be a probability that the computer will fool the human into believing that it is human. Turing thus displaces the question of artificial intelligence towards its natural conceptual horizon of probability, where truth is the provisory production of a set of rules. Artificial intelligence is not a matter of

[2]A French translation of Water Trier's *8192 Crazy People in One Book For Children from 5 and under to 75 and over*, where 30 illustrations of stereotypical characters are cut in three parts (head, body, legs) and spiral-bound so as to offer 8192 potential combinations.

whether computers can think, Turing argues, it is a matter of establishing the probability that a computer can fool a human, or more precisely, that a human can be fooled by a machine.

Turing then answers the most common objections to the concept of artificial intelligence by performing conceptual displacements across what Wittgenstein called games of language. For example, one of these objections is that computers could never write a sonnet on command, which Turing's machine elegantly, and almost humanly, answers, "Count me out on this one. I never could write poetry." Turing's quotation performs a similar trick by displacing the question of artificial intelligence towards the provisory truth constructed in a specific communicational situation. Through Turing's quote, Queneau suggests that a reader could momentarily suspend their disbelief towards machines, project themself on a fictionalized computerized reader, and interpret a computer-generated sonnet as poetry. In other words, a computer can write a poetical sonnet, on the condition that there is at least one reader willing to pretend it is. Thus the question is not whether a machine can be struck by poetical inspiration, but whether a reader can read a machine-generated sonnet as a poem through the fiction of an ideal reader turned into an ideal computer-reader. The significance of Turing's epigraph in Queneau's *CMMP* is not to be interpreted as a warning against the lack of humanity in artificial intelligence, nor is it the expression of the belief in a computer mind. Instead, Queneau uses Turing's powerful philosophical intuition as a way to displace reading from the realm of truth to the realm of probability, and the question of textuality from production to reception. This difference is important to literature, because it displaces randomness from the field of revelation, inner truth, and writing constraints, to the field of communication, reader-response, and game.

Starynkevitch's Program

The same year of its print publication, a first digital version of *CMMP* was programmed by D. Starynkevitch on a CAB 500 (Bens 2005). We don't know much about it other that Queneau received poems allegedly created by Starynkevitch's program. The response of the Oulipo to Starynkevitch's poems reflects mistrust against randomness:

> We wished that M. Starynkevitch would explain the method used; we hoped that the choice of the verses was not left to randomness. (Bens 2005, 79)[3]

What was the reason for this negative reaction? Some argue (Bloomfield and Campaignolle 2016) that it is because of the Oulipo's suspicion of randomness. But that would be confusing two very different forms of randomness: pure randomness as used by the surrealists and pseudo-randomness as used by computer scientists within perfectly describable programs. There were enough mathematicians and information scientists amongst the Oulipo to differentiate between the surrealists' use of pure randomness and the benefits of pseudo-randomness for potential literature.

Pseudo-random numbers are a series of numbers completely determined by the combination of a given digit, called a seed or a key, and a specific algorithm. As such, this series can be reproduced, but its logic is so complex that it is considered to simulate true randomness, even though it should really be called complexity, and not randomness. Pseudo-random number generators have always been a core part of most, if not all, programming language, because pseudo-randomness is essential to the calculus of probabilities: in order to test the probability, for instance, of a missile to launch properly in spite of wind and other unpredictable elements, it is necessary to test a program under a form of randomness that can be reproduced. Thus, pseudo-randomness in computing is very different from randomness as in "a dice cup". It might look the same, but you cannot reproduce the outcome of a dice cup. Computer-generated randomness is unique in the sense that, unlike dice cups, it is used to sound probabilities.

It is true that the Oulipo built its aesthetic program partly against randomness. Jacques Bens explains the Oulipo's dislike for randomness in colorful ways in 1968:

> Oulipo members never concealed their hatred for randomness, pity fortune-tellers and cheap haphazardness: "The Oulipo is the anti-randomness," once said Claude Berge very seriously, "which leaves no doubt about how much we loath dice cups […]. Potentiality is uncertain, but not random. We perfectly know everything

[3] "On souhaita que M. Starynkevitch nous précise la méthode utilisé ; on espéra que le choix des vers ne fut pas laissé au hasard".

> that can possibly happen, we just don't know if it will." (Bens 1968)

This moral stand against randomness relies mainly on a mistrust towards the surrealists' use of randomness in automatic writing. In particular, the Oulipo was reacting to the association between automatic writing and the free exploration of the unconscious promoted by surrealists. The Oulipian attack on randomness is based on the idea that automatic writing only frees the writer from formal constraints but subjects him to unconscious writing reflexes. On the contrary, the use of consciously self-imposed constraints serves as a lever for a freer form of literary creation, because it pushes writers to step away from their writing reflexes. But no valid argument is given by Jacques Bens against the specific use of dice cups. In fact, Queneau's position is much more nuanced than Bens' and Berge's:

> We are perhaps not so very "anti" [-randomness]. I would prefer to say that we display a certain suspicion with regard to chance. (Bens 2005, 146)[4]

Oulipians were aware of the fact that the Oulipo creates constraints that can have a great degree of arbitrariness, if not randomness, in its process[5]. A constraint like S+7 is no less arbitrary than a constraint like the "exquisite corpse", and both can be described as "automatic". The essential difference between the surrealist's and the Oulipian's use of arbitrariness is the latter only leverages it in constraints that can be perfectly described. Only a constraint of which the structure can be perfectly described has potential, because it can be reused many times under similar circumstances. Automatic writing, in that sense, cannot be perfectly described and is not an Oulipian constraint.

But what if one could create a computer program that generates automatic writing under perfectly describable rules? This program was written and executed in 1964 by Jean Baudot in his *La Machine à écrire*. Commenting on this work (Baudot 1964, 81), Queneau describes the outcome of Baudot's machine as "automatic texts", which he compares to "surrealist sentences". Queneau seems enthusiastic enough to suggest him that he creates an imita-

[4] "Nous ne sommes peut-être pas tellement 'anti'. Je préfèrerais dire que nous manifestons une certaine méfiance à l'égard du hasard."

[5] See the discussion about "automatism", "randomness", and "structure" in Bens (2005).

tion game inspired by Turing's famous imitation game. In Queneau's version of that game, different testers would be presented with "surrealist sentences" composed by humans, and sentences composed by Baudot's *Machine à écrire*. At the end of the game, testers would have to identify the origin of each sentence. Queneau's interest in such a game helps understand how he could appreciate experiments in automatic text generation as a contribution to automatic writing in a way that departed from Surrealism. Randomness becomes a site of Oulipian potentiality if it is used in a computer program that is a perfectly describable structure. If surrealist automatic writing is not Oulipian, computer-generated surrealist writing definitely is.

Randomness, Humanisms, and Automata

Paul Braffort's 1975 version of Queneau's *CMMP* showed that computer randomness is in fact extremely useful to the exploration of potentiality, especially in the case of *CMMP*. Paul Fournel states that a digital version of *CMMP* with a pseudo-random function can be used to sound the potential outcomes of *CMMP*:

> The author himself can take advantage of such an edition: when the combinations are so numerous, he can proceed to controls by soundings. In this case the computer has the role of an assistant to the edition of the definitive text. (Fournel 1981, 299)[6]

Pseudo-random numbers are thus essential to the exploration of complex forms of potentiality. This use of randomness echoes Turing's discussion of the ability of machines to learn and the need to use randomness to achieve this goal. Turing compares different methods through which a computer could scan an array of potential solutions in order to find the correct ones by itself. Randomness, Turing argues, is the most efficient way for a machine to find correct solutions among an array of answers, because there is a better chance that solutions are positioned randomly on that array rather than in close groups:

[6] "L'auteur lui-même peut faire son profit d'une telle édition : lorsque les combinaisons sont aussi nombreuses, il peut procéder à des contrôles par sondage. L'ordinateur joue dans ce cas un rôle d'assistant à la mise au point définitive du texte."

> A random element is rather useful when we are searching for a solution of some problem. Suppose for instance we wanted to find a number between 50 and 200 which was equal to the square of the sum of its digits, we might start at 51 then try 52 and go on until we got a number that worked. Alternatively we might choose numbers at random until we got a good one. […] The systematic method has the disadvantage that there may be an enormous block without any solutions in the region which has to be investigated first […]. (Turing 1950, 459)

Since there is probably a very large number of satisfactory solutions, the random method seems to be better than the systematic. So whether it is used to look for solutions or errors, pseudo-randomness is essential to the exploration of complex systems like *CMMP*. This is probably why pseudo-randomness was consistently used in all digital versions of *CMMP*. Yet, my hypothesis is that there is a tension at stake in the use of pseudo-randomness to sound the potentiality of constraints. As Turing illustrates, the use of pseudo-randomness to sound potentiality brings us very close to the concept of a self-learning machine. It is precisely this overlap that authors and scholars of generative poetry pick on when they present *CMMP* as a form of proto-text generators, rather than as a form of hypertext. But this overlap might be problematic for the Oulipian aesthetics, which are based on a strong sense of bond between the author and the reader. Dubbed an aesthetic of bonding by Hervé Le Tellier (2006), it seems contradictory with the idea of a text generated without any kind of human interaction by a self-learning machine. Paul Fournel neutralizes this overlap when he limits randomness to being an instrument for sounding, a tool to assist the author or the reader in their editing and reading of the text. In 1981, Paul Fournel and other Oulipians created the ALAMO, the workshop for mathematic-and-computer-assisted literature, whose mission is to use computers to sound the potentiality of constrained literature. The notion of computer-assistance used by the ALAMO can be interpreted, in my opinion, as a way to neutralize the tension that exists between the concept of artificial intelligence and that of constrained literature. The ALAMO presents readers with the image of a computer that is only there to assist humans in reading or creating combinatory works, not to take their place.

Human and Computers: Beings of Letters

However, this tension resurfaces in any digital version of *CMMP*, and my hypothesis is that a dissonance between artificial intelligence and the Oulipian aesthetic of bonding explains the Oulipo's initial disquiet when facing Starynkevitch's first digital version of *CMMP*. Paul Braffort's version of *CMMP* adds to pseudo-randomness a new feature that seems to have the sole function of creating the fictional character of an Oulipian, conniving reader within the machine, a ghost in the shell. In his digital version of *CMMP,* Braffort chose to seed a pseudo-random numbers generator with a digit calculated on the basis of the letters of the user's name as well as the time it took that user to enter it in the program[7]. The program still uses pseudo-randomness, but in a way that makes the user feel like they are part of the process. At the end of the process, the poem is signed by both Queneau's and the user's name. Braffort's original solution is to inject the computer with something from the reader. In that sense, Braffort creates a new instance of this fictional character that the Oulipo keeps creating: the Oulipian reader. Braffort creates some sort of cyborg-Oulipian-reader.

Entering one's name in the terminal seems reminiscent of two forms of Oulipian constraints. It recalls the Oulipian constraint known as "beau présent", where one writes a text using only the letters of the dedicatee's name. It also reminds of the way Noël Arnaud uses computer language in his "Poèmes Algol". The constraint was originally to compose poems with words from a computer language known as Algorithmic Oriented Language. But Noël Arnaud, performing what Oulipians call a "clinamen", loosens up this original constraint by using *letters* from words composing the Algorithmic Oriented Language, which enables him to write much richer poems (Oulipo 1973). As in Paul Braffort's use of readers' names to seed the computer-generated random function, the decomposition of words into letters seems to be a common way to reintroduce humanity into a literature reacting to the context of computing. Even though the user's configurational function is replaced by pseudo-randomness in Braffort's version of *CMMP*, Braffort manages to preserve the Oulipian aesthetics of bonding by playing on an algebraic interpretation of the letters composing one's name. But maybe Braffort is

[7] "L'ordinateur, lui, opère une sélection dans le corpus à partir de la longueur du nom du 'lecteur' et du temps qu'il met à le dactylographier sur le terminal puis édite le sonnet qui porte la double signature de Queneau et de son lecteur" (Fournel 1981, 299).

in fact introducing computing into humanity by interpreting human names as a series of digits. Indeed, François Le Lionnais was developing poems of punctuation as Braffort was developping his version of the *CMMP*. The problem of code is the same one as the problem of punctuation, as John Cayley noticed: punctuation doesn't have meaning but programs a scripted performance, like "pause", "raise your voice", etc. In other words, punctuation is what humans have in common with computers. I interpret Braffort's gesture to introduce the user's name into the program as a way to suggest that, no matter how "post-human" computers are, and no matter how disquieting the use of pseudo-randomness is for a humanistic view of literature, there is a strong bond between computing and humanness. This bond exists at the level of signifiers that, like characters, digits, and punctuation, while scafolding human speech, do not produce meaning as much as they perform the repetition of a programmed task. The very materiality, the very texture of language, its letters and punctuation, is what humans have in common with computers. This might be a good reason to better appreciate a sonnet written by another machine.

Bibliography

Baudot, Jean A. 1964. *La Machine à écrire, mise en marche et programmée par Jean A. Baudot. Le Premier recueil de vers libres rédigés par un ordinateur électronique*. Montréal: Les éditions du jour.

Bens, Jacques. 1968. "Littérature potentielle." *L'Arc*, no. 28.

———. 2005. *Genèse de l'Oulipo : 1960-1963. Circulaire du 28 août 1961*. Le Pré-Saint-Gervais: Le Castor Astral.

Bloomfield, Camille, and Hélène Campaignolle. 2016. "Machines littéraires, machines numériques : l'Oulipo et l'informatique." In *Oulipo mode d'emploi*, edited by C. Reggiani and A. Schnaffner, 319–36. Paris: Honoré Champion éditeur.

Fournel, Paul. 1981. "Ordinateur et écrivain : l'expérience du Centre Pompidou." In *Atlas de littérature potentielle*. Paris: Gallimard.

Le Tellier, Hervé. 2006. *Esthétique de l'Oulipo*. Le Pré-Saint-Gervais: Le Castor Astral.

Oulipo. 1973. *La Littérature potentielle*. Paris: Gallimard.

Queneau, Raymond. 1961. "Mode d'emploi." In *Cent mille milliard de poèmes*. Paris: Gallimard.

Turing, A.M. 1950. "Computing Machinery and Intelligence." *Mind* 59 (236): 433–60.