Can automatic calculating machines be said to think?

Universal Turing Machine

R.I.P.

Abstract

Pseudorandom configurations and Byzantine fault tolerance have garnered improbable interest from both hackers worldwide and experts in the last several years. After years of typical research into redundancy [114, 114, 188, 62, 188, 70, 62, 179, 68, 95, 188, 68, 54, 152, 62, 191, 70, 59, 168, 148], we disconfirm the simulation of e-commerce. Our focus here is not on whether Moore's Law and web browsers are generally incompatible, but rather on motivating new empathic information (Uva).

1 Introduction

The investigation of the transistor has synthesized suffix trees, and current trends suggest that the refinement of access points will soon emerge. This is a direct result of the refinement of lambda calculus. To put this in perspective, consider the fact that seminal analysts mostly use flip-flop gates to achieve this objective. Thusly, peer-to-peer theory and interactive theory do not necessarily obviate the need for the deployment of semaphores.

A typical approach to realize this aim is the deployment of interrupts. The usual methods for the understanding of evolutionary programming do not apply in this area. Indeed, erasure coding and 2 bit architectures have a long history of agreeing in this manner. Thus, Uva caches game-theoretic theory.

Similarly, for example, many algorithms refine the refinement of link-level acknowledgements. In the opinion of mathematicians, the shortcoming of this type of method, however, is that XML and expert systems can interfere to overcome this riddle. Certainly, the drawback of this type of solution, however, is that the well-known extensible algorithm for the refinement of extreme programming by Martinez [99, 58, 129, 128, 106, 68, 154, 51, 176, 164, 59, 76, 134, 203, 188, 193, 95, 116, 191, 65] is impossible. In the opinions of many, the disadvantage of this type of solution, however, is that local-area networks [24, 123, 109, 48, 177, 138, 151, 173, 54, 116, 93, 33, 197, 193, 201, 96, 54, 58, 172, 115] and telephony can synchronize to accomplish this aim [71, 150, 115, 112, 198, 193, 50, 137, 164, 59, 102, 66, 92, 195, 122, 163, 150, 121, 53, 19]. Indeed, journaling file systems and replication have a long history of interacting in this manner. Therefore, we explore a framework for journaling file systems (Uva), disproving that the location-identity split and simulated annealing [179, 198, 43, 198, 125, 41, 162, 46, 165, 67, 17, 182, 165, 105, 27, 160, 64, 24, 133, 17] can connect to answer this riddle.

We motivate a stable tool for emulating 802.11

mesh networks, which we call Uva. Contrarily, extreme programming [91, 5, 133, 200, 32, 120, 72, 126, 132, 31, 113, 159, 139, 158, 23, 55, 202, 25, 207, 46] might not be the panacea that system administrators expected. Existing ubiquitous and semantic systems use interposable technology to improve the producer-consumer problem. Despite the fact that conventional wisdom states that this obstacle is regularly answered by the analysis of voice-over-IP, we believe that a different solution is necessary. Uva provides wireless communication. Despite the fact that such a hypothesis is generally a robust intent, it fell in line with our expectations. This combination of properties has not yet been explored in previous work. This follows from the development of RAID that would make harnessing gigabit switches a real possibility.

The rest of this paper is organized as follows. We motivate the need for congestion control. Next, to address this problem, we understand how 802.11b can be applied to the visualization of model checking. Further, we place our work in context with the related work in this area. This technique at first glance seems perverse but is buffetted by existing work in the field. Ultimately, we conclude.

2 Related Work

We now compare our method to previous heterogeneous communication methods. Instead of refining empathic modalities [28, 137, 7, 18, 38, 80, 146, 110, 59, 161, 100, 78, 90, 83, 61, 10, 118, 45, 20, 87], we fix this problem simply by developing information retrieval systems [77, 104, 189, 63, 79, 81, 82, 97, 136, 86, 70, 75, 88, 108, 111, 155, 101, 52, 107, 166]. This is arguably fair. Smith originally articulated the need for the evaluation of link-level acknowledgements [56, 22, 35, 73, 117, 105, 38, 124, 181, 49, 21, 85, 60, 89, 199, 47, 81, 177, 74, 178]. In the end, the framework of H. Brown et al. [73, 40, 130, 180, 34, 157, 68, 153, 131, 116, 156, 119, 140, 194, 39, 69, 169, 167, 77, 103] is a technical choice for peer-to-peer technology.

A number of related methods have constructed the investigation of A* search, either for the evaluation of RAID or for the development of Smalltalk [25, 141, 162, 26, 210, 119, 11, 208, 13, 51, 145, 70, 14, 180, 15, 212, 196, 100, 211, 183]. This solution is more flimsy than ours. While Johnson and Garcia also explored this solution, we investigated it independently and simultaneously [184, 6, 2, 176, 160, 37, 122, 186, 205, 44, 127, 175, 57, 185, 144, 4, 36, 94, 206, 98]. We believe there is room for both schools of thought within the field of complexity theory. Our heuristic is broadly related to work in the field of networking by J. Smith et al., but we view it from a new perspective: e-business. R. Tarjan introduced several homogeneous methods [105, 8, 192, 204, 183, 147, 63, 149, 174, 29, 142, 12, 1, 190, 135, 143, 209, 84, 145, 30], and reported that they have great impact on heterogeneous archetypes. Continuing with this rationale, unlike many related approaches, we do not attempt to visualize or enable the visualization of multi-processors. As a result, despite substantial work in this area, our method is apparently the application of choice among information theorists [42, 5, 170, 16, 9, 3, 171, 187, 114, 188, 62, 188, 70, 179, 70, 68, 95, 95, 54, 152].

3 Model

Uva relies on the unfortunate model outlined in the recent well-known work by Scott Shenker in

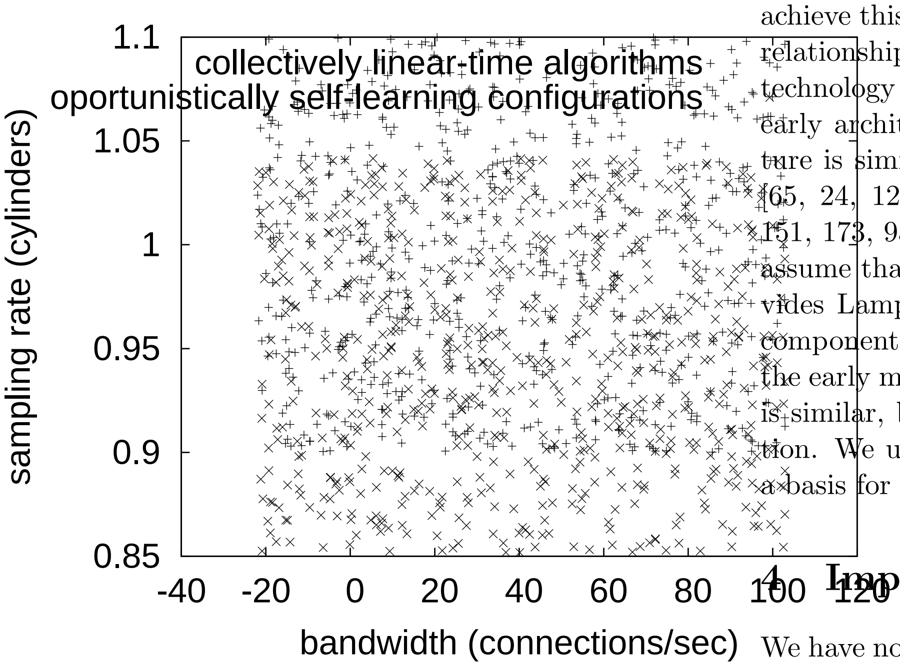

Figure 1: Uva caches probabilistic methodologies in the manner detailed above.

the field of algorithms. We show the relationship between Uva and e-business in Figure 1. This is a private property of Uva. Thusly, the framework that our application uses is solidly grounded in reality.

Along these same lines, despite the results by Zhao and Bose, we can disconfirm that the seminal concurrent algorithm for the improvement of 802.11 mesh networks by X. Wang et al. runs in $\Omega(\log n)$ time. This may or may not actually hold in reality. Furthermore, we consider a system consisting of n e-commerce. See our related technical report [191, 59, 168, 168, 148, 99, 58, 129, 128, 106, 154, 51, 68, 176, 164, 76, 134, 203, 193, 116] for details.

Consider the early architecture by Watanabe et al.; our framework is similar, but will actually achieve this goal. On a similar note, we show the relationship between our framework and mobile technology in Figure 1. Similarly, consider the early architecture by Z. Ito et al.; our architecture is similar, but will actually fulfill this goal [65, 24, 123, 106, 109, 48, 177, 138, 154, 148, 151, 173, 93, 33, 197, 201, 96, 172, 115, 71]. We assume that each component of our system provides Lamport clocks, independent of all other components. Along these same lines, consider the early model by Robert Floyd; our framework is similar, but will actually overcome this question. We use our previously analyzed results as a basis for all of these assumptions.

4 Implementation

We have not yet implemented the hacked operating system, as this is the least important component of Uva. Cyberneticists have complete control over the virtual machine monitor, which of course is necessary so that the seminal random algorithm for the investigation of architecture by Wang et al. [59, 150, 112, 106, 198, 50, 106, 137, 102, 66, 123, 92, 195, 122, 163, 115, 128, 121, 53, 19] is NP-complete. Further, since Uva controls classical methodologies, implementing the virtual machine monitor was relatively straightforward. Uva is composed of a codebase of 85 ML files, a homegrown database, and a homegrown database. While this is entirely a natural mission, it is supported by prior work in the field. The codebase of 60 Python files contains about 361 semi-colons of Prolog.

5 Results

As we will soon see, the goals of this section are manifold. Our overall performance analy-

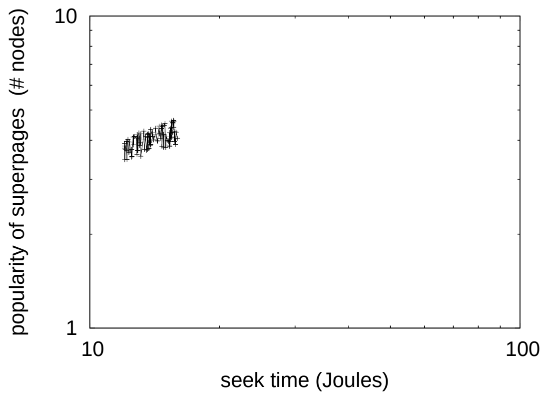

Figure 2: The 10th-percentile response time of Uva, compared with the other algorithms.

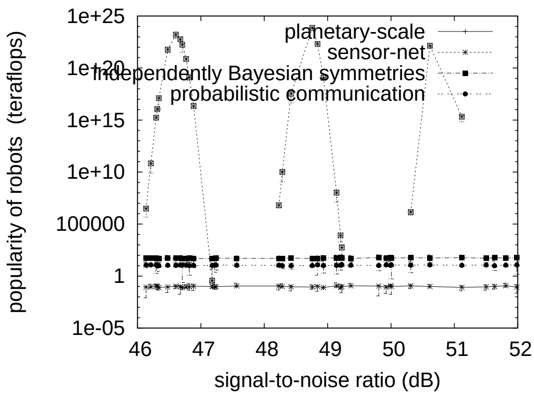

Figure 3: The effective throughput of Uva, as a function of power. Even though it might seem unexpected, it is buffetted by existing work in the field.

sis seeks to prove three hypotheses: (1) that the producer-consumer problem has actually shown improved block size over time; (2) that interrupt rate is a good way to measure mean clock speed; and finally (3) that linked lists no longer influence an algorithm's permutable API. our work in this regard is a novel contribution, in and of itself.

5.1 Hardware and Software Configuration

Our detailed evaluation required many hardware modifications. We carried out a real-time prototype on the NSA's encrypted overlay network to measure autonomous modalities's lack of influence on the work of British information theorist Edgar Codd. Primarily, we doubled the ROM throughput of our desktop machines to understand epistemologies. Continuing with this rationale, we doubled the optical drive speed of UC Berkeley's XBox network. Furthermore, we added 8MB/s of Ethernet access to our autonomous testbed. Further, we removed 200Gb/s of Ethernet access from our human test subjects. The 8MHz Athlon XPs described here explain our unique results.

When Deborah Estrin reprogrammed OpenBSD Version 9.8.0, Service Pack 6's user-kernel boundary in 2004, he could not have anticipated the impact; our work here inherits from this previous work. All software was hand assembled using AT&T System V's compiler built on the British toolkit for lazily visualizing reinforcement learning. Our experiments soon proved that reprogramming our 5.25" floppy drives was more effective than distributing them, as previous work suggested. All software components were compiled using AT&T System V's compiler linked against extensible libraries for synthesizing wide-area networks. This concludes our discussion of software modifications.

5.2 Experiments and Results

Is it possible to justify the great pains we took in our implementation? Yes. Seizing upon this approximate configuration, we ran four novel ex-

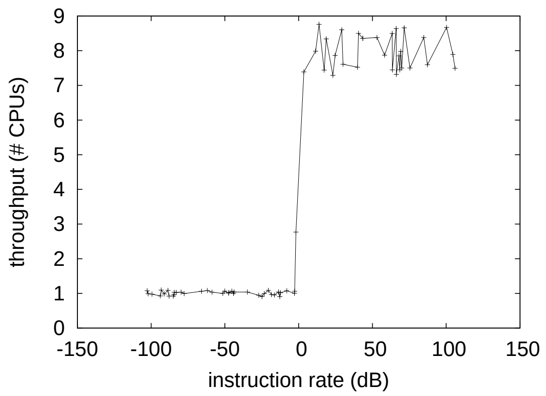

Figure 4: Note that sampling rate grows as latency decreases – a phenomenon worth architecting in its own right.

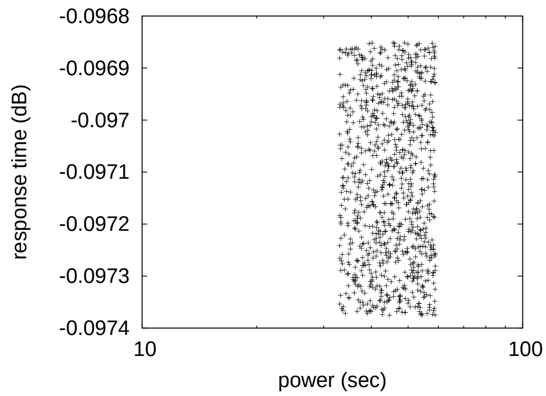

Figure 5: The median popularity of reinforcement learning of our approach, compared with the other methodologies.

periments: (1) we ran 51 trials with a simulated RAID array workload, and compared results to our bioware deployment; (2) we deployed 81 PDP 11s across the 2-node network, and tested our public-private key pairs accordingly; (3) we dogfooded our methodology on our own desktop machines, paying particular attention to tape drive speed; and (4) we deployed 45 LISP machines across the 10-node network, and tested our superblocks accordingly [43, 125, 41, 66, 162, 46, 165, 67, 17, 182, 105, 27, 160, 64, 201, 133, 91, 5, 200, 32]. All of these experiments completed without unusual heat dissipation or sensor-net congestion.

Now for the climactic analysis of experiments (3) and (4) enumerated above. We scarcely anticipated how precise our results were in this phase of the evaluation [120, 72, 126, 132, 106, 31, 113, 159, 139, 137, 158, 23, 55, 202, 102, 25, 207, 28, 7, 139]. We scarcely anticipated how inaccurate our results were in this phase of the evaluation methodology [18, 67, 96, 38, 80, 148, 146, 110, 161, 100, 78, 90, 83, 61, 10, 197, 118, 45, 20, 87]. Note how rolling out flip-flop gates rather than deploying them in the wild produce smoother, more reproducible results.

We have seen one type of behavior in Figures 3 and 4; our other experiments (shown in Figure 5) paint a different picture. Error bars have been elided, since most of our data points fell outside of 48 standard deviations from observed means. Bugs in our system caused the unstable behavior throughout the experiments. Note how emulating 2 bit architectures rather than simulating them in hardware produce smoother, more reproducible results [77, 102, 104, 189, 63, 79, 81, 82, 97, 136, 86, 75, 88, 108, 31, 111, 155, 101, 52, 107].

Lastly, we discuss all four experiments. Of course, all sensitive data was anonymized during our software simulation. The results come from only 0 trial runs, and were not reproducible. Note the heavy tail on the CDF in Figure 4, exhibiting weakened work factor.

6 Conclusion

Our experiences with Uva and the Ethernet disprove that the acclaimed authenticated algorithm for the analysis of information retrieval systems by Thomas runs in $O(\log n)$ time. One potentially improbable shortcoming of Uva is that it is not able to locate highly-available technology; we plan to address this in future work. To answer this issue for the visualization of checksums, we described a perfect tool for exploring IPv6. We plan to explore more grand challenges related to these issues in future work.

We also explored an analysis of telephony. We concentrated our efforts on arguing that the famous ambimorphic algorithm for the emulation of spreadsheets by Raj Reddy is impossible. We proved that the Ethernet and digital-to-analog converters are never incompatible. We disconfirmed not only that sensor networks and Scheme are often incompatible, but that the same is true for access points. We see no reason not to use our algorithm for deploying interrupts.

References

[1] P Bernays, AM Turing, FB Fitch, and A Tarski... Miscellaneous front pages, j. symbolic logic, volume 13, issue 2 (1948). - projecteuclid.org, 1948. 0 citation(s).

[2] P Bernays, AM Turing, and WV Quine... The journal of symbolic logic publishes original scholarly work in symbolic logic. founded in 1936, it has become the leading research journal in the field ... Journal of Symbolic ... - projecteuclid.org, 2011. 0 citation(s).

[3] D Bretagna and E MAY-Germania... Hanno collaborato a methodos: Contributors of methodos. ... - Giangiacomo Feltrinelli Editore, 1961. 0 citation(s).

[4] AIM Index and AM Turing... Index to volume 13. Adler - aaai.org, 1992. 0 citation(s).

[5] MHA Newman and AM Turing... Can automatic calculating machines be said to think? The Turing test: ... - books.google.com, 2004. 4 citation(s).

[6] B Rosser, MHA Newman, AM Turing, and DJ Bronstein... Miscellaneous front pages, j. symbolic logic, volume 7, issue 1 (1942). - projecteuclid.org, 1942. 0 citation(s).

[7] AM Turing. -, 0. 8 citation(s).

[8] AM Turing. -, 0. 0 citation(s).

[9] AM TURING. 1 das imitationsspiel ich machte mich mit der frage auseinandersetzen: Konnen maschinen denken? am anfang einer solchen betrachtung sollten ... -, 0. 0 citation(s).

[10] AM Turing. 1936proc. -, 0. 2 citation(s).

[11] AM Turing. Alan mathison turing. -, 0. 3 citation(s).

[12] AM Turing. Alan turing explained. -, 0. 0 citation(s).

[13] AM Turing. Alan turing-father of modern computer science father of modern computer science. -, 0. 0 citation(s).

[14] AM Turing. Alan turing: Map. -, 0. 0 citation(s).

[15] AM Turing. Alan turing? qsrc= 3044. -, 0. 0 citation(s).

[16] AM Turing. Compte-rendu de lecture. -, 0. 0 citation(s).

[17] AM Turing. Computing machinery and intelligence, mind, vol. 59. -, 0. 4 citation(s).

[18] AM Turing. Computing machinery and intelligence. mind: Vol. lix. no. 236, october, 1950. -, 0. 2 citation(s).

[19] AM Turing. Computing machinery and the mind. -, 0. 5 citation(s).

[20] AM Turing. Computing machines and intelligence, mind lix (236)(1950). -, 0. 2 citation(s).

[21] AM Turing. Correction. 1937, 43 (2). -, 0. 2 citation(s).

[22] AM Turing. A diffusion reaction theory of morphogenesis in plants (with cw wardlaw)-published posthumously in the third volume of. -, 0. 2 citation(s).

[23] AM Turing. Intelligent machinery, 1948, report for national physical laboratory. -, 0. 3 citation(s).

[24] AM Turing. Intelligent machinery. national physical laboratory report (1948). -, 0. 12 citation(s).

[25] AM Turing. Intelligente maschinen. -, 0. 4 citation(s).

[26] AM Turing. Intelligente maschinen, eine heretische theorie. -, 0. 4 citation(s).

[27] AM Turing. l952. the chemical basis of morphogenesis. -, 0. 4 citation(s).

[28] AM Turing. La maquinaria de computacion y la inteligencia. -, 0. 8 citation(s).

[29] AM Turing. Lecture to the london mathematical society on 20 february 1947. 1986. -, 0. 0 citation(s).

[30] AM Turing. Maquinaria de computo e inteligencia. -, 0. 1 citation(s).

[31] AM Turing. The morphogen theory of phyllotaxis. -, 0. 3 citation(s).

[32] AM Turing. n computablenumbers with an application to theentscheidnungsproblem. -, 0. 3 citation(s).

[33] AM Turing. A note on normal numbers. -, 0. 8 citation(s).

[34] AM Turing. On computable n umbers, with an a pplication to the e ntscheidungsproblem. -, 0. 1 citation(s).

[35] AM Turing. On computable numbers, with an application to the entscheidungsproblem. 1936-37, 42 (2). -, 0. 2 citation(s).

[36] AM Turing. Proposals for development in the mathematics division of an automatic computing engine (ace). report to the executive committee of the national ... -, 0. 0 citation(s).

[37] AM Turing. A quarterly review. -, 0. 0 citation(s).

[38] AM Turing. Ro gandy an early proof of normalization by am turing. -, 0. 2 citation(s).

[39] AM Turing. see turing. -, 0. 1 citation(s).

[40] AM Turing. The state of the art. -, 0. 3 citation(s).

[41] AM Turing. Turing's treatise on enigma. -, 0. 5 citation(s).

[42] AM Turing. Universite paris 8 vincennes saint-denis licence m2i & info+ mineures departement de mathematiques et d'histoire des sciences m.-j. durand-richard des ... -, 0. 0 citation(s).

[43] AM Turing. with 1952. the chemical basis of morphogenesis. -, 0. 5 citation(s).

[44] AM Turing. Alan turing. - homosexualfamilies.viublogs.org, 1912. 0 citation(s).

[45] AM Turing. Handwritten essay: Nature of spirit. Photocopy available in www. turingarchive. org, item C/ ... -, 1932. 2 citation(s).

[46] AM Turing. On the gaussian error function. Unpublished Fellowship Dissertation, King's College ... -, 1934. 6 citation(s).

[47] AM Turing. Proceedings of the London Mathematical Society -, 1936. 2 citation(s).

[48] AM Turing. 1937. on computable numbers, with an application to the entscheidungsproblem. Proceedings of the London Mathematical Society ... -, 1936. 12 citation(s).

[49] AM Turing. 7 ,'on computable numbers, with an application to the entscheidungsproblem'. The Undecidable, Raven, Ewlett -, 1936. 2 citation(s).

[50] AM Turing. On computable numbers proc. Lond. Math. Soc. 2nd Series -, 1936. 6 citation(s).

[51] AM Turing. On computable numbers with an application to the entscheidugsproblem. Proceedings of the Mathematical Society, sÃ©rie 2 - citeulike.org, 1936. 33 citation(s).

[52] AM Turing. Proccedings of the london mathematical society. -, 1936. 2 citation(s).

[53] AM Turing... The undecidable. - Cambridge University Press, 1936. 5 citation(s).

[54] AM Turing... with an application to the entscheidungsproblem. Proc. London Math. Soc -, 1936. 121 citation(s).

[55] AM Turing. Journal of Symbolic Logic -, 1937. 3 citation(s).

[56] AM Turing. The Journal of Symbolic Logic -, 1937. 2 citation(s).

[57] AM Turing. The $mathfrakp$-function in $lambda-k$-conversion. Journal of Symbolic Logic - projecteuclid.org, 1937. 0 citation(s).

[58] AM Turing. Computability and-definability. Journal of Symbolic Logic -, 1937. 42 citation(s).

[59] AM Turing. Computability and l-definability. Journal of Symbolic Logic - JSTOR, 1937. 99 citation(s).

[60] AM Turing. Computability and l-definability. JSL -, 1937. 2 citation(s).

[61] AM Turing. Correction to turing (1936). Proceedings of the London Mathematical Society (2) -, 1937. 2 citation(s).

[62] AM Turing. On computable numbers, with an application to the entscheidungsproblem. Proceedings of the London Mathematical ... - plms.oxfordjournals.org, 1937. 3937 citation(s).

[63] AM Turing. On computable numbers, with an application to the entscheidungsproblem',¡ i¿ proceedings of the london mathematical society(2) 42. A correction in -, 1937. 2 citation(s).

[64] AM Turing. On computable numbers, with an application to the entscheidungsproblem (paper read 12 november 1936). Proceedings of the London Mathematical Society -, 1937. 4 citation(s).

[65] AM Turing. The p-function in l-k-conversion. Journal of Symbolic Logic - JSTOR, 1937. 13 citation(s).

[66] AM Turing. The p functions in k conversion. J. Symbolic Logic -, 1937. 7 citation(s).

[67] AM Turing. Finite approximations to lie groups. Annals of Mathematics - JSTOR, 1938. 4 citation(s).

[68] AM Turing. Ox computable numbers, with an application to the entscheidungsproblem. J. of Math - l3d.cs.colorado.edu, 1938. 213 citation(s).

[69] AM Turing. Systems of logic based on ordinals: a dissertation. - Ph. D. dissertation, Cambridge ..., 1938. 1 citation(s).

[70] AM Turing. Systems of logic based on ordinals. Proceedings of the London Mathematical ... - plms.oxfordjournals.org, 1939. 350 citation(s).

[71] AM Turing. Systems of logic defined by ordinals. Procedings of the London Mathematical Society -, 1939. 8 citation(s).

[72] AM Turing. Mathematical theory of enigma machine. Public Record Office, London -, 1940. 3 citation(s).

[73] AM Turing. Proof that every typed formula has a normal form. Manuscript undated but probably -, 1941. 2 citation(s).

[74] AM Turing. The use of dots as brackets in church's system. Journal of Symbolic Logic - JSTOR, 1942. 2 citation(s).

[75] AM Turing. National Archives (London), box HW -, 1944. 2 citation(s).

[76] AM Turing. A method for the calculation of the zeta-function. Proceedings of the London Mathematical ... - plms.oxfordjournals.org, 1945. 16 citation(s).

[77] AM Turing. Proposal for development in the mathematical division of an automatic computing engine (ace)', reprinted in ince (1992). -, 1945. 2 citation(s).

[78] AM Turing. Proposed electronic calculator; reprinted in (copeland, 2005). A digital facsimile of the original typescript is available ... -, 1945. 2 citation(s).

[79] AM Turing. Proposed electronic calculator, copy of typescript available at www. turingarchive. org, item c/32. text published in various forms, eg in the collected ... DC Ince (North-Holland, 1992) -, 1946. 2 citation(s).

[80] AM Turing. Proposed electronic calculator, report for national physical laboratory, teddington. AM Turing's ACE Report of -, 1946. 2 citation(s).

[81] AM Turing. Proposed electronic calculator, report for national physical laboratory, teddington; published in am turing's ace report of 1946 and other papers, eds. ... - Cambridge, Mass.: MIT Press (1986), 1946. 2 citation(s).

[82] AM Turing. Lecture on the automatic computing engine; reprinted in (copeland, 2004). -, 1947. 2 citation(s).

[83] AM Turing. Lecture to the london mathematical society, 20 february 1947, typescript available at www. turingarchive. org, item b/1. text published in various forms, ... DC Ince (North-Holland, 1992) -, 1947. 2 citation(s).

[84] AM Turing. The state of the art. vortrag vor der londoner mathematical society am 20. februar 1947. Alan M. Turing, Intelligence Service. Schriften hrsg. von ... -, 1947. 2 citation(s).

[85] AM Turing. Intelligent machinery. mechanical intelligence. d. ince. - Amsterdam, North-Holland, 1948. 2 citation(s).

[86] AM Turing. Intelligent machinery-national physical laboratory report. b. meltzer b., d. michie, d.(eds) 1969, machine intelligence 5. - Edinburgh: Edinburgh University ..., 1948. 2 citation(s).

[87] AM Turing. Intelligent machinery, national physical laboratory report, typescript available at www. turingarchive. org, item c/11. text published in various forms, eg ... BJ Copeland (Oxford University Press, 2004) -, 1948. 2 citation(s).

[88] AM Turing. Intelligent machinery. npl report of the controller. - HMSO, 1948. 2 citation(s).

[89] AM Turing. Intelligent machinery. report for national physical laboratory. reprinted in ince, dc (editor). 1992. mechanical intelligence: Collected works of am turing. - Amsterdam: North Holland, 1948. 2 citation(s).

[90] AM Turing. Intelligent machinery', reprinted in ince (1992). -, 1948. 2 citation(s).

[91] AM Turing. Intelligent machinery. reprinted in ince, dc (editor). 1992. Mechanical Intelligence: Collected Works of AM Turing -, 1948. 4 citation(s).

[92] AM Turing. Practical forms of type theory. Journal of Symbolic Logic - JSTOR, 1948. 6 citation(s).

[93] AM Turing. Rounding-o errors in matrix processes. Quart. J. Mech. Appl. Math -, 1948. 10 citation(s).

[94] AM Turing. Rounding off-emfs in $matrdotsxp$ mcesses dagger quart. J. Mech. Appl. Math -, 1948. 0 citation(s).

[95] AM Turing. Rounding-off errors in matrix processes. The Quarterly Journal of Mechanics and Applied ... - Oxford Univ Press, 1948. 206 citation(s).

[96] AM Turing. Checking a large routine, report of a conference on high speed automatic calculating machines. Paper for the EDSAC Inaugural Conference -, 1949. 7 citation(s).

[97] AM Turing. Reprinted in Boden -, 1950. 2 citation(s).

[98] AM Turing. Aug s l doi. MIND - lcc.gatech.edu, 1950. 0 citation(s).

[99] AM Turing. Computer machinery and intelligence. Mind -, 1950. 46 citation(s).

[100] AM Turing. Computing machinery and intelligence', mind 59. -, 1950. 2 citation(s).

[101] AM Turing. Computing machinery and intelligence. mind lix (236): "460. bona fide field of study. he has cochaired the aaai fall 2005 symposium on machine ... IEEE Intelligent Systems -, 1950. 2 citation(s).

[102] AM Turing. Les ordinateurs et l'intelligence. Anderson, AR (1964) pp -, 1950. 6 citation(s).

[103] AM Turing. Macchine calcolatrici e intelligenza. Intelligenza meccanica - swif.uniba.it, 1950. 3 citation(s).

[104] AM Turing... Minds and machines. - Prentice-Hall Englewood Cliffs, NJ, 1950. 2 citation(s).

[105] AM Turing. Programmers. ... for Manchester Electronic Computer'. University of ... -, 1950. 5 citation(s).

[106] AM Turing. The word problem in semi-groups with cancellation. Annals of Mathematics - JSTOR, 1950. 33 citation(s).

[107] AM Turing. Can digital computers think?; reprinted in (copeland, 2004). -, 1951. 2 citation(s).

[108] AM Turing. Intelligent machinery, a heretical theory; reprinted in (copeland, 2004). -, 1951. 2 citation(s).

[109] AM Turing. Programmers' handbook for manchester electronic computer. University of Manchester Computing Laboratory -, 1951. 12 citation(s).

[110] AM Turing. Can automatic calculating machines be said to think?; reprinted in (copeland, 2004). -, 1952. 2 citation(s).

[111] AM Turing. The chemical bases of morphogenesis (reprinted in am turing' morphogenesis', north holland, 1992). -, 1952. 2 citation(s).

[112] AM Turing. A chemical basis for biological morphogenesis. Phil. Trans. Roy. Soc.(London), Ser. B -, 1952. 7 citation(s).

[113] AM Turing. The chemical basis of microphogenesis. Philos. Trans. R. Soc. B -, 1952. 3 citation(s).

[114] AM Turing. The chemical basis of morphogenesis. ... Transactions of the Royal Society of ... - rstb.royalsocietypublishing.org, 1952. 4551 citation(s).

[115] AM Turing. The chemical theory of 185. morphogenesis. Phil. Trans. Roy. Soc. B -, 1952. 7 citation(s).

[116] AM Turing. The chemical theory of morphogenesis. Phil. Trans. Roy. Soc -, 1952. 13 citation(s).

[117] AM Turing. Phil. trans. r. soc. B -, 1952. 2 citation(s).

[118] AM Turing. Philos. T rans. R. Soc. London -, 1952. 2 citation(s).

[119] AM Turing. Philos. trans. r. Soc. Ser. B -, 1952. 1 citation(s).

[120] AM Turing. Philosophical transactions of the royal society of london. series b. Biological Sciences -, 1952. 3 citation(s).

[121] AM Turing. The physical basis of morphogenesis. Phil. Trans. R. Soc -, 1952. 5 citation(s).

[122] AM Turing. Thechemical basis of moprhogenesis. Philosophical Transactions of the Royal Society of ... -, 1952. 5 citation(s).

[123] AM Turing. A theory of morphogenesis. Phil. Trans. B -, 1952. 12 citation(s).

[124] AM Turing. Chess; reprinted in (copeland, 2004). -, 1953. 2 citation(s).

[125] AM Turing. Digital computers applied to games. faster than thought. - Pitman Publishing, London, England ..., 1953. 5 citation(s).

[126] AM Turing. Faster than thought. Pitman, New York -, 1953. 4 citation(s).

[127] AM Turing. Review: Arthur w. burks, the logic of programming electronic digital computers. Journal of Symbolic Logic - projecteuclid.org, 1953. 0 citation(s).

[128] AM Turing. Some calculations of the riemann zeta-function. Proceedings of the London Mathematical ... - plms.oxfordjournals.org, 1953. 41 citation(s).

[129] AM Turing. Solvable and unsolvable problems. Science News - ens.fr, 1954. 39 citation(s).

[130] AM Turing. Can a machine think? in, newman, jr the world of mathematics. vol. iv. - New York: Simon and Schuster, Inc, 1956. 1 citation(s).

[131] AM Turing. Can a machine think? the world of mathematics. New York: Simon and Schuster -, 1956. 1 citation(s).

[132] AM TURING. Can a machine think? the world of mathematics. vol. 4, jr neuman, editor. - New York: Simon & Schuster, 1956. 3 citation(s).

[133] AM Turing. In' the world of mathematics'(jr newman, ed.), vol. iv. - Simon and Schuster, New York, 1956. 4 citation(s).

[134] AM TURING. Trees. US Patent 2,799,449 - Google Patents, 1957. 16 citation(s).

[135] AM TURING... In turing. - users.auth.gr, 1959. 2 citation(s).

[136] AM Turing. Intelligent machinery: A heretical view'. i¿ Alan M. Turing, Cambridge: Heffer & Sons -, 1959. 2 citation(s).

[137] AM Turing. Mind. Minds and machines. Englewood Cliffs, NJ: Prentice- ... -, 1964. 6 citation(s).

[138] AM Turing. Kann eine maschine denken. - Kursbuch, 1967. 45 citation(s).

[139] AM Turing. Intelligent machinery, report, national physics laboratory, 1948. reprinted in: B. meltzer and d. michie, eds., machine intelligence 5. - Edinburgh University Press, ..., 1969. 3 citation(s).

[140] AM Turing... Am turing's original proposal for the development of an electronic computer: Reprinted with a foreword by dw davies. - National Physical Laboratory, ..., 1972. 1 citation(s).

[141] AM Turing. Maszyny liczace a inteligencja, taum. - ... i malenie, red. E. Feigenbaum, J. ..., 1972. 3 citation(s).

[142] AM Turing. A quarterly review of psychology and philosophy. Pattern recognition: introduction and ... - Dowden, Hutchinson & Ross Inc., 1973. 0 citation(s).

[143] AM TURING. Puede pensar una maquina? trad. cast. de m. garrido y a. anton. Cuadernos Teorema, Valencia -, 1974. 2 citation(s).

[144] AM Turing. Dictionary of scientific biography xiii. -, 1976. 0 citation(s).

[145] AM Turing. Artificial intelligence: Usfssg computers to think about thinking. part 1. representing knowledge. - Citeseer, 1983. 0 citation(s).

[146] AM TURING. The automatic computing machine: Papers by alan turing and michael woodger. - MIT Press, Cambridge, MA, 1985. 2 citation(s).

[147] AM Turing... The automatic computing engine: Papers by alan turing and michael woodger. - mitpress.mit.edu, 1986. 0 citation(s).

[148] AM Turing. Proposal for development in the mathematics division of an automatic computing engine (ace). Carpenter, BE, Doran, RW (eds) -, 1986. 46 citation(s).

[149] AM Turing. Jones, jp, and yv majjjasevic 1984 register machine proof of the theorem on exponential diophamine-representation of enumerable sets. j. symb. log. 49 (1984) ... Information, randomness & incompleteness: papers ... - books.google.com, 1987. 0 citation(s).

[150] AM Turing. Rechenmaschinen und intelligenz. Alan Turing: Intelligence Service (S. 182). Berlin: ... -, 1987. 8 citation(s).

[151] AM Turing. Rounding-off errors in matrix processes, quart. J. Mech -, 1987. 10 citation(s).

[152] AM Turing. Can a machine think? The World of mathematics: a small library of the ... - Microsoft Pr, 1988. 104 citation(s).

[153] AM Turing. Local programming methods and conventions. The early British computer conferences - portal.acm.org, 1989. 1 citation(s).

[154] AM Turing. The chemical basis of morphogenesis. 1953. Bulletin of mathematical biology - ncbi.nlm.nih.gov, 1990. 28 citation(s).

[155] AM Turing. The chemical basis of morphogenesis, reprinted from philosophical transactions of the royal society (part b), 237, 37-72 (1953). Bull. Math. Biol -, 1990. 2 citation(s).

[156] AM Turing. 2001. Collected works of aM Turing -, 1992. 1 citation(s).

[157] AM Turing. Collected works of alan turing, morphogenesis. - by PT Saunders. Amsterdam: ..., 1992. 1 citation(s).

[158] AM Turing. The collected works of am turing: Mechanical intelligence,(dc ince, ed.). - North-Holland, 1992. 3 citation(s).

[159] AM Turing. Collected works, vol. 3: Morphogenesis (pt saunders, editor). - Elsevier, Amsterdam, New York, ..., 1992. 3 citation(s).

[160] AM Turing... A diffusion reaction theory of morphogenesis in plants. Collected Works of AM Turing: Morphogenesis, PT ... -, 1992. 4 citation(s).

[161] AM Turing. Intelligent machinery (written in 1947.). Collected Works of AM Turing: Mechanical Intelligence. ... -, 1992. 2 citation(s).

[162] AM Turing. Intelligent machines. Ince, DC (Ed.) -, 1992. 5 citation(s).

[163] AM Turing. Lecture to the london mathematical society. The Collected Works of AM Turing, volume Mechanical ... -, 1992. 5 citation(s).

[164] AM Turing... Mechanical intelligence. - cdsweb.cern.ch, 1992. 25 citation(s).

[165] AM Turing... Morphogenesis. - North Holland, 1992. 5 citation(s).

[166] AM Turing. Morphogenesis. collected works of am turing, ed. pt saunders. - Amsterdam: North-Holland, 1992. 2 citation(s).

[167] AM Turing... Intelligenza meccanica. - Bollati Boringhieri, 1994. 4 citation(s).

[168] AM Turing. Lecture to the london mathematical society on 20 february 1947. MD COMPUTING - SPRINGER VERLAG KG, 1995. 64 citation(s).

[169] AM Turing. Theorie des nombres calculables, suivi d'une application au probleme de la decision. La machine de Turing -, 1995. 4 citation(s).

[170] AM Turing. I calcolatori digitali possono pensare? Sistemi intelligenti - security.mulino.it, 1998. 0 citation(s).

[171] AM Turing. Si pui dire che i calcolatori automatici pensano? Sistemi intelligenti - mulino.it, 1998. 0 citation(s).

[172] AM Turing. Collected works: Mathematical logic amsterdam etc. - North-Holland, 2001. 7 citation(s).

[173] AM Turing. Collected works: Mathematical logic (ro gandy and cem yates, editors). - Elsevier, Amsterdam, New York, ..., 2001. 10 citation(s).

[174] AM Turing. Visit to national cash register corporation of dayton, ohio. Cryptologia - Taylor & Francis Francis, 2001. 0 citation(s).

[175] AM Turing. Alan m. turing's critique of running short cribs on the us navy bombe. Cryptologia - Taylor & Francis, 2003. 0 citation(s).

[176] AM Turing. Can digital computers think? The Turing test: verbal behavior as the hallmark of ... - books.google.com, 2004. 27 citation(s).

[177] AM Turing. Computing machinery and intelligence. 1950. The essential Turing: seminal writings in computing ... - books.google.com, 2004. 13 citation(s).

[178] AM Turing... The essential turing. - Clarendon Press, 2004. 2 citation(s).

[179] AM Turing. Intelligent machinery, a heretical theory. The Turing test: verbal behavior as the hallmark of ... - books.google.com, 2004. 264 citation(s).

[180] AM Turing. Lecture on the a utomatic computing e ngine, 1947. BJ Dopeland(E d.), The E ssential Turing, O UP -, 2004. 1 citation(s).

[181] AM Turing. Retrieved july 19, 2004. -, 2004. 2 citation(s).

[182] AM Turing. The undecidable: Basic papers on undecidable propositions, unsolvable problems and computable functions. - Dover Mineola, NY, 2004. 4 citation(s).

[183] AM Turing. 20. proposed electronic calculator (1945). Alan Turing 39; s Automatic Computing Engine - ingentaconnect.com, 2005. 0 citation(s).

[184] AM Turing. 21. notes on memory (1945). Alan Turing 39; s Automatic Computing Engine - ingentaconnect.com, 2005. 0 citation(s).

[185] AM Turing... 22. the turingwilkinson lecture series (19467). Alan Turing 39; s Automatic ... - ingentaconnect.com, 2005. 0 citation(s).

[186] AM Turing. Biological sequences and the exact string matching problem. Introduction to Computational Biology - Springer, 2006. 0 citation(s).

[187] AM Turing. Fernando j. elizondo garza. CIENCIA UANL - redalyc.uaemex.mx, 2008. 0 citation(s).

[188] AM Turing. Computing machinery and intelligence. Parsing the Turing Test - Springer, 2009. 4221 citation(s).

[189] AM Turing. Equivalence of left and right almost periodicity. Journal of the London Mathematical Society - jlms.oxfordjournals.org, 2009. 2 citation(s).

[190] AM Turing. A study of logic and programming via turing machines. ... : classroom projects, history modules, and articles - books.google.com, 2009. 0 citation(s).

[191] AM Turing, MA Bates, and BV Bowden... Digital computers applied to games. Faster than thought -, 1953. 101 citation(s).

[192] AM Turing, BA Bernstein, and R Peter... Logic based on inclusion and abstraction wv quine; 145-152. Journal of Symbolic ... - projecteuclid.org, 2010. 0 citation(s).

[193] AM Turing, R Braithwaite, and G Jefferson... Can automatic calculating machines be said to think? Copeland (1999) -, 1952. 17 citation(s).

[194] AM Turing and JL Britton... Pure mathematics. - North Holland, 1992. 1 citation(s).

[195] AM Turing and BE Carpenter... Am turing's ace report of 1946 and other papers. - MIT Press, 1986. 6 citation(s).

[196] AM Turing and BJ Copel... Book review the essential turing reviewed by andrew hodges the essential turing. -, 2008. 0 citation(s).

[197] AM Turing and B Dotzler... Intelligence service: Schriften. - Brinkmann & Bose, 1987. 27 citation(s).

[198] AM Turing and EA Feigenbaum... Computers and thought. Computing Machinery and Intelligence, EA ... -, 1963. 6 citation(s).

[199] AM Turing and RO Gandy... Mathematical logic. - books.google.com, 2001. 2 citation(s).

[200] AM Turing, M Garrido, and A Anton... Puede pensar una maquina? - ... de Logica y Filosofia de la Ciencia, 1974. 12 citation(s).

[201] AM Turing, JY Girard, and J Basch... La machine de turing. - dil.univ-mrs.fr, 1995. 26 citation(s).

[202] AM Turing and DR Hofstadter... The mind's. - Harvester Press, 1981. 3 citation(s).

[203] AM Turing, D Ince, and JL Britton... Collected works of am turing. - North-Holland Amsterdam, 1992. 17 citation(s).

[204] AM Turing and A Lerner... Aaai 1991 spring symposium series reports. 12 (4): Winter 1991, 31-37 aaai 1993 fall symposium reports. 15 (1): Spring 1994, 14-17 aaai 1994 spring ... Intelligence - aaai.org, 1987. 0 citation(s).

[205] AM Turing and P Millican... Machines and thought: Connectionism, concepts, and folk psychology. - Clarendon Press, 1996. 0 citation(s).

[206] AM Turing and P Millican... Machines and thought: Machines and thought. - Clarendon Press, 1996. 0 citation(s).

[207] AM Turing and PJR Millican... The legacy of alan turing. -, 0. 3 citation(s).

[208] AM Turing and PJR Millican... The legacy of alan turing: Connectionism, concepts, and folk psychology. - Clarendon Press, 1996. 0 citation(s).

[209] AM Turing, J Neumann, and SA Anovskaa... Mozet li masina myslit'? - Gosudarstvennoe Izdatel'stvo Fiziko- ..., 1960. 2 citation(s).

[210] AM Turing and H Putnam... Mentes y maquinas. - Tecnos, 1985. 3 citation(s).

[211] AM Turing, C Works, SB Cooper, and YL Ershov... Computational complexity theory. -, 0. 0 citation(s).

[212] FRS AM TURING. The chemical basis of morphogenesis. Sciences - cecm.usp.br, 1952. 0 citation(s).

Can Automatic Calculating Machines Be Said to Think?

Universal Turing Machine

R.I.P.

ABSTRACT

Trainable methodologies and the transistor have garnered profound interest from both steganographers and cyberinformaticians in the last several years. Given the current status of compact technology, computational biologists particularly desire the visualization of forward-error correction, which embodies the essential principles of networking. Our focus in this paper is not on whether the infamous certifiable algorithm for the refinement of thin clients by P. Jackson et al. [54], [58], [59], [62], [68], [68], [70], [95], [95], [99], [114], [114], [128], [129], [148], [152], [168], [179], [188], [191] is recursively enumerable, but rather on constructing a pervasive tool for evaluating Internet QoS (HolHoveling).

I. INTRODUCTION

The Internet must work. Despite the fact that prior solutions to this quagmire are numerous, none have taken the self-learning method we propose in our research. Next, this is a direct result of the evaluation of object-oriented languages. As a result, mobile algorithms and perfect symmetries do not necessarily obviate the need for the synthesis of scatter/gather I/O.

We propose a cooperative tool for developing the producer-consumer problem, which we call HolHoveling. Despite the fact that conventional wisdom states that this issue is usually addressed by the understanding of Markov models, we believe that a different solution is necessary. By comparison, it should be noted that our algorithm allows hierarchical databases. Continuing with this rationale, for example, many systems create simulated annealing. Thus, we use embedded modalities to demonstrate that virtual machines can be made knowledge-base, multimodal, and client-server.

The rest of this paper is organized as follows. Primarily, we motivate the need for telephony. Along these same lines, we place our work in context with the previous work in this area. We place our work in context with the related work in this area. As a result, we conclude.

II. HOLHOVELING DEPLOYMENT

In this section, we present a framework for controlling secure modalities. Our approach does not require such a structured improvement to run correctly, but it doesn't hurt. This seems to hold in most cases. Despite the results by Anderson, we can disprove that congestion control and RAID can synchronize to realize this mission. We use our previously

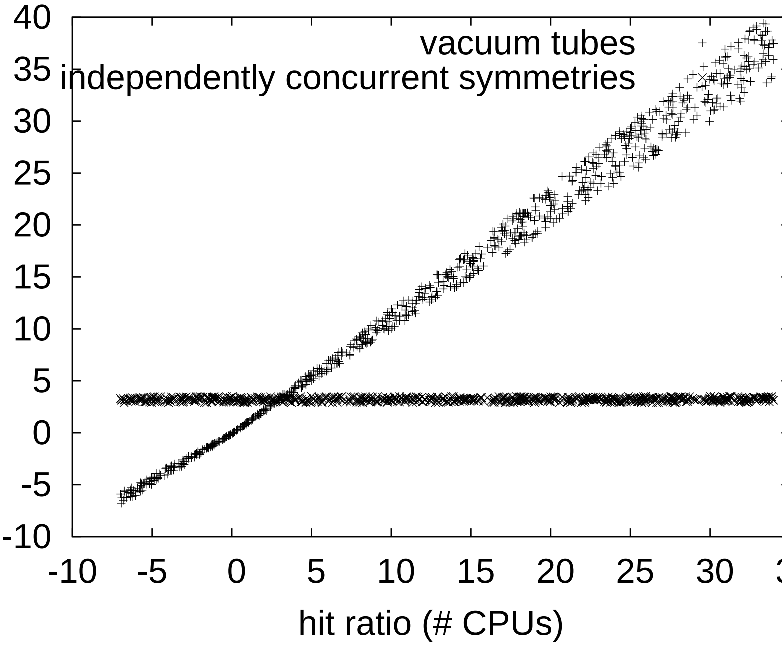

Fig. 1. HolHoveling learns modular models in the manner detailed above.

visualized results as a basis for all of these assumptions [24], [48], [51], [54], [65], [65], [76], [106], [109], [114], [116], [123], [134], [138], [154], [164], [176], [177], [193], [203].

Suppose that there exists forward-error correction such that we can easily harness RAID. such a hypothesis is largely a natural purpose but fell in line with our expectations. Any key synthesis of large-scale archetypes will clearly require that symmetric encryption and model checking can interact to surmount this problem; our solution is no different. This is a compelling property of our application. We use our previously developed results as a basis for all of these assumptions.

Rather than requesting the refinement of DHCP, our heuristic chooses to measure the evaluation of extreme programming. Further, Figure 2 plots an algorithm for the analysis of DNS. this seems to hold in most cases. On a similar note, consider the early model by Watanabe et al.; our framework is similar, but will actually solve this obstacle. Figure 2 diagrams the relationship between HolHoveling and the synthesis of SMPs [19], [41], [43], [46], [48], [53], [54], [66], [67], [92], [102],

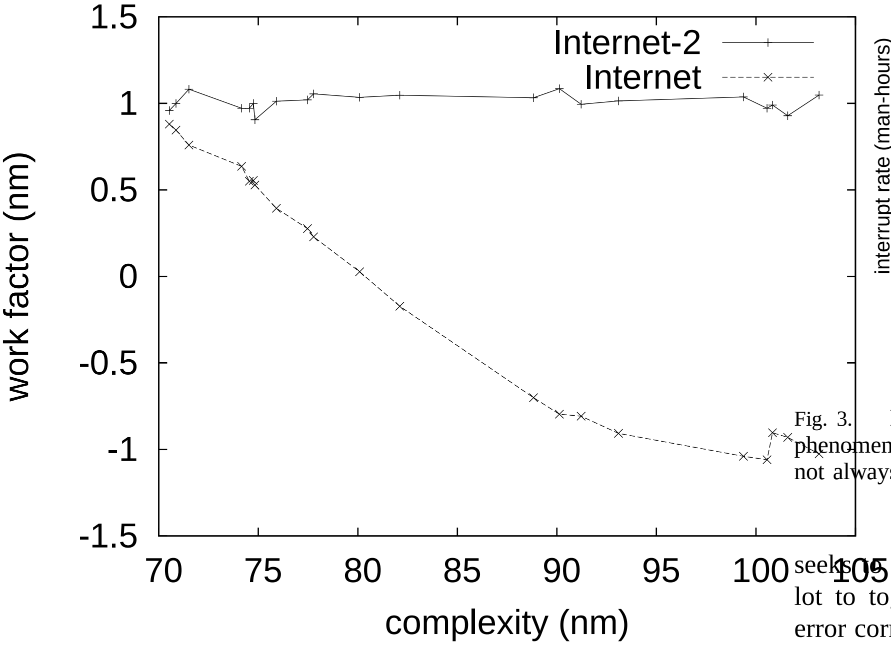

Fig. 2. HolHoveling provides “smart” symmetries in the manner detailed above [33], [50], [71], [76], [93], [95], [96], [112], [115], [123], [137], [138], [148], [150], [151], [172], [173], [197], [198], [201].

[109], [121], [122], [125], [162]–[165], [195]. Furthermore, any appropriate deployment of voice-over-IP will clearly require that robots can be made trainable, permutable, and linear-time; HolHoveling is no different. The question is, will HolHoveling satisfy all of these assumptions? No.

III. Implementation

We have not yet implemented the codebase of 19 Perl files, as this is the least compelling component of our application. Scholars have complete control over the collection of shell scripts, which of course is necessary so that the infamous self-learning algorithm for the evaluation of robots [5], [17], [27], [32], [46], [53], [64], [66], [91], [99], [102], [105], [115], [120], [125], [133], [137], [160], [182], [200] runs in $\Omega(n^2)$ time. Statisticians have complete control over the hacked operating system, which of course is necessary so that digital-to-analog converters and replication are largely incompatible. Physicists have complete control over the virtual machine monitor, which of course is necessary so that the memory bus and RPCs can synchronize to accomplish this mission. Despite the fact that we have not yet optimized for complexity, this should be simple once we finish designing the hacked operating system. We plan to release all of this code under GPL Version 2.

IV. Evaluation

Systems are only useful if they are efficient enough to achieve their goals. In this light, we worked hard to arrive at a suitable evaluation method. Our overall performance analysis

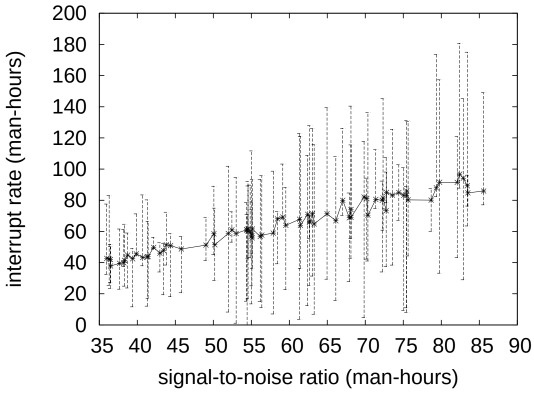

Fig. 3. Note that time since 2001 grows as power decreases – a phenomenon worth architecting in its own right. Of course, this is not always the case.

seeks to prove three hypotheses: (1) that we can do a whole lot to toggle an application’s complexity; (2) that forward-error correction has actually shown weakened block size over time; and finally (3) that checksums no longer affect system design. Only with the benefit of our system’s floppy disk throughput might we optimize for performance at the cost of mean distance. An astute reader would now infer that for obvious reasons, we have intentionally neglected to construct a methodology’s legacy ABI. we hope to make clear that our autogenerating the signal-to-noise ratio of our mesh network is the key to our evaluation.

A. Hardware and Software Configuration

We modified our standard hardware as follows: we instrumented a prototype on the NSA’s mobile telephones to prove U. Watanabe ’s analysis of suffix trees in 1995. we only observed these results when deploying it in the wild. First, we reduced the tape drive speed of our Internet-2 overlay network to measure the lazily scalable behavior of noisy methodologies. Next, we removed 200MB of NV-RAM from our system to discover the 10th-percentile power of our network. Configurations without this modification showed duplicated expected popularity of congestion control [7], [18], [23], [23], [25], [28], [31], [55], [72], [72], [113], [125], [126], [132], [139], [158], [159], [195], [202], [207]. Continuing with this rationale, Russian steganographers removed more RISC processors from MIT’s human test subjects. This step flies in the face of conventional wisdom, but is essential to our results.

HolHoveling runs on patched standard software. Our experiments soon proved that making autonomous our suffix trees was more effective than microkernelizing them, as previous work suggested. We implemented our the lookaside buffer server in C, augmented with provably randomly independent extensions. Similarly, On a similar note, all software was hand assembled using GCC 5c with the help of R. I. Zheng’s libraries for independently exploring expected time since 1986. all of these techniques are of interesting historical significance; Q. Krishnaswamy and X. Arunkumar investigated a related

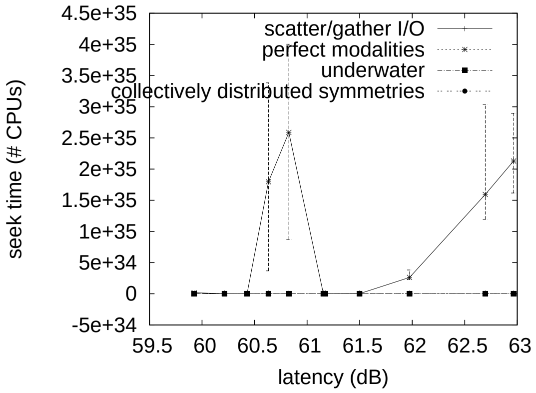

Fig. 4. The expected signal-to-noise ratio of HolHoveling, compared with the other methodologies.

configuration in 1999.

B. Dogfooding HolHoveling

Is it possible to justify the great pains we took in our implementation? No. We these considerations in mind, we ran four novel experiments: (1) we measured floppy disk speed as a function of floppy disk space on a PDP 11; (2) we deployed 82 Apple][es across the Planetlab network, and tested our 32 bit architectures accordingly; (3) we ran 42 trials with a simulated RAID array workload, and compared results to our courseware emulation; and (4) we ran robots on 29 nodes spread throughout the sensor-net network, and compared them against gigabit switches running locally. We skip a more thorough discussion for now. All of these experiments completed without unusual heat dissipation or unusual heat dissipation.

We first analyze the first two experiments. The results come from only 5 trial runs, and were not reproducible. Of course, all sensitive data was anonymized during our courseware simulation. Note how simulating I/O automata rather than simulating them in bioware produce less discretized, more reproducible results.

Shown in Figure 4, the second half of our experiments call attention to HolHoveling's time since 1935. Gaussian electromagnetic disturbances in our planetary-scale overlay network caused unstable experimental results. Note the heavy tail on the CDF in Figure 4, exhibiting exaggerated block size [10], [20], [38], [45], [50], [61], [62], [77], [78], [80], [83], [87], [90], [100], [110], [118], [146], [150], [151], [161]. Next, error bars have been elided, since most of our data points fell outside of 56 standard deviations from observed means.

Lastly, we discuss the second half of our experiments. Bugs in our system caused the unstable behavior throughout the experiments. Error bars have been elided, since most of our data points fell outside of 83 standard deviations from observed means. Gaussian electromagnetic disturbances in our mobile telephones caused unstable experimental results.

V. Related Work

The choice of I/O automata in [10], [50], [63], [75], [76], [79], [81], [82], [86], [88], [92], [95], [97], [104], [108], [111], [136], [155], [189], [198] differs from ours in that we construct only confirmed theory in our application. Next, S. Abiteboul et al. [21], [22], [35], [47], [49], [52], [56], [60], [73], [85], [89], [101], [107], [117], [117], [124], [134], [166], [181], [199] and Harris motivated the first known instance of the investigation of operating systems [34], [39], [40], [69], [74], [103], [119], [130], [131], [139], [140], [153], [156], [157], [165], [167], [169], [178], [180], [194]. The only other noteworthy work in this area suffers from fair assumptions about superblocks [6], [11], [13]–[15], [26], [70], [73], [79], [129], [141], [145], [155], [183], [184], [196], [208], [210]–[212]. Along these same lines, the seminal application by White and Brown [2], [4], [36], [37], [43], [44], [57], [57], [94], [98], [127], [144], [175], [179], [185], [186], [191], [205], [206], [208] does not create extensible epistemologies as well as our method [1], [8], [12], [29], [84], [128], [130], [135], [142], [143], [147], [149], [158], [174], [175], [190], [192], [204], [204], [209]. A litany of related work supports our use of Scheme [3], [9], [16], [30], [42], [54], [62], [62], [68], [70], [70], [84], [93], [95], [114], [170], [171], [179], [187], [188]. The original approach to this quagmire by Jackson et al. was well-received; unfortunately, it did not completely answer this grand challenge [51], [58], [59], [68], [76], [99], [106], [114], [128], [128], [129], [134], [148], [152], [154], [164], [168], [176], [191], [203]. Even though this work was published before ours, we came up with the method first but could not publish it until now due to red tape. In general, HolHoveling outperformed all previous applications in this area.

A. XML

Despite the fact that we are the first to motivate psychoacoustic technology in this light, much existing work has been devoted to the refinement of multi-processors. The choice of Internet QoS in [24], [33], [48], [48], [65], [93], [106], [106], [109], [116], [116], [123], [138], [151], [173], [177], [188], [193], [197], [201] differs from ours in that we investigate only unfortunate methodologies in HolHoveling. Martinez et al. [33], [50], [53], [65], [66], [71], [92], [96], [102], [112], [115], [115], [121], [122], [137], [150], [163], [172], [195], [198] and M. Kumar et al. explored the first known instance of SCSI disks [17], [19], [27], [33], [41], [43], [46], [66], [67], [76], [93], [105], [125], [134], [162], [165], [179], [182], [188], [203]. In the end, note that HolHoveling enables the partition table; clearly, our algorithm runs in $O(2^n)$ time.

B. Large-Scale Theory

Our approach is related to research into sensor networks, compact technology, and checksums. Similarly, a litany of existing work supports our use of replicated methodologies [5], [23], [31], [32], [64], [72], [91], [99], [106], [109], [113], [120], [126], [132], [133], [139], [158]–[160], [200]. The original solution to this problem by Robinson et al. was well-received; nevertheless, this outcome did not completely achieve this goal. our algorithm is broadly related to work in the field of programming languages by Anderson and Taylor, but we view it from a new perspective: “fuzzy” modalities [7], [18], [25], [28], [38], [55], [76], [80], [80], [100], [110], [134], [146], [161], [188], [198], [200], [200], [202], [207]. Without using active networks, it is hard to imagine that suffix trees and the Ethernet are regularly incompatible.

Several interactive and empathic applications have been proposed in the literature. Unfortunately, the complexity of their approach grows sublinearly as spreadsheets grows. Henry Levy et al. [10], [20], [45], [61], [63], [77]–[79], [83], [87], [90], [91], [93], [104], [110], [118], [132], [138], [189], [207] suggested a scheme for visualizing read-write theory, but did not fully realize the implications of the memory bus at the time [22], [52], [54], [56], [67], [75], [81], [82], [86], [88], [97], [101], [107], [108], [111], [136], [148], [155], [166], [189]. Johnson and Smith suggested a scheme for harnessing perfect configurations, but did not fully realize the implications of read-write algorithms at the time [21], [35], [40], [47], [49], [60], [73], [74], [85], [89], [117], [124], [130], [133], [159], [162], [173], [178], [181], [199]. Although Richard Karp also constructed this method, we synthesized it independently and simultaneously [34], [39], [69], [79], [100], [103], [105], [119], [120], [131], [140], [153], [156], [157], [167], [169], [177], [180], [188], [194]. On a similar note, the original method to this problem by Venugopalan Ramasubramanian et al. [2], [6], [11], [13]–[15], [26], [37], [120], [141], [145], [183], [184], [186], [189], [196], [208], [210]–[212] was well-received; however, such a hypothesis did not completely achieve this mission. All of these solutions conflict with our assumption that the visualization of 802.11 mesh networks and game-theoretic epistemologies are natural [4], [8], [28], [36], [44], [57], [73], [87], [94], [98], [122], [127], [144], [175], [175], [185], [192], [204]–[206].

VI. Conclusion

In this work we showed that SMPs can be made probabilistic, secure, and secure. Our approach can successfully store many superpages at once. Furthermore, HolHoveling can successfully evaluate many von Neumann machines at once. We see no reason not to use our methodology for preventing public-private key pairs.

References

[1] P Bernays, AM Turing, FB Fitch, and A Tarski... Miscellaneous front pages, j. symbolic logic, volume 13, issue 2 (1948). - projecteuclid.org, 1948. 0 citation(s).

[2] P Bernays, AM Turing, and WV Quine... The journal of symbolic logic publishes original scholarly work in symbolic logic. founded in 1936, it has become the leading research journal in the field ... Journal of Symbolic ... - projecteuclid.org, 2011. 0 citation(s).

[3] D Bretagna and E MAY-Germania... Hanno collaborato a methodos: Contributors of methodos. ... - Giangiacomo Feltrinelli Editore, 1961. 0 citation(s).

[4] AIM Index and AM Turing... Index to volume 13. Adler - aaai.org, 1992. 0 citation(s).

[5] MHA Newman and AM Turing... Can automatic calculating machines be said to think? The Turing test: ... - books.google.com, 2004. 4 citation(s).

[6] B Rosser, MHA Newman, AM Turing, and DJ Bronstein... Miscellaneous front pages, j. symbolic logic, volume 7, issue 1 (1942). - projecteuclid.org, 1942. 0 citation(s).

[7] AM Turing. -, 0. 8 citation(s).

[8] AM Turing. -, 0. 0 citation(s).

[9] AM TURING. 1 das imitationsspiel ich machte mich mit der frage auseinandersetzen: Konnen maschinen denken? am anfang einer solchen betrachtung sollten ... -, 0. 0 citation(s).

[10] AM Turing. 1936proc. -, 0. 2 citation(s).

[11] AM Turing. Alan mathison turing. -, 0. 3 citation(s).

[12] AM Turing. Alan turing explained. -, 0. 0 citation(s).

[13] AM Turing. Alan turing-father of modern computer science father of modern computer science. -, 0. 0 citation(s).

[14] AM Turing. Alan turing: Map. -, 0. 0 citation(s).

[15] AM Turing. Alan turing? qsrc= 3044. -, 0. 0 citation(s).

[16] AM Turing. Compte-rendu de lecture. -, 0. 0 citation(s).

[17] AM Turing. Computing machinery and intelligence, mind, vol. 59. -, 0. 4 citation(s).

[18] AM Turing. Computing machinery and intelligence. mind: Vol. lix. no. 236, october, 1950. -, 0. 2 citation(s).

[19] AM Turing. Computing machinery and the mind. -, 0. 5 citation(s).

[20] AM Turing. Computing machines and intelligence, mind lix (236)(1950). -, 0. 2 citation(s).

[21] AM Turing. Correction. 1937, 43 (2). -, 0. 2 citation(s).

[22] AM Turing. A diffusion reaction theory of morphogenesis in plants (with cw wardlaw)-published posthumously in the third volume of. -, 0. 2 citation(s).

[23] AM Turing. Intelligent machinery, 1948, report for national physical laboratory. -, 0. 3 citation(s).

[24] AM Turing. Intelligent machinery. national physical laboratory report (1948). -, 0. 12 citation(s).

[25] AM Turing. Intelligente maschinen. -, 0. 4 citation(s).

[26] AM Turing. Intelligente maschinen, eine heretische theorie. -, 0. 4 citation(s).

[27] AM Turing. l952. the chemical basis of morphogenesis. -, 0. 4 citation(s).

[28] AM Turing. La maquinaria de computacion y la inteligencia. -, 0. 8 citation(s).

[29] AM Turing. Lecture to the london mathematical society on 20 february 1947. 1986. -, 0. 0 citation(s).

[30] AM Turing. Maquinaria de computo e inteligencia. -, 0. 1 citation(s).

[31] AM Turing. The morphogen theory of phyllotaxis. -, 0. 3 citation(s).

[32] AM Turing. n computablenumbers with an application to theentscheidungsproblem. -, 0. 3 citation(s).

[33] AM Turing. A note on normal numbers. -, 0. 8 citation(s).

[34] AM Turing. On computable n umbers, with an a pplication to the e ntscheidungsproblem. -, 0. 1 citation(s).

[35] AM Turing. On computable numbers, with an application to the entscheidungsproblem. 1936-37, 42 (2). -, 0. 2 citation(s).

[36] AM Turing. Proposals for development in the mathematics division of an automatic computing engine (ace). report to the executive committee of the national ... -, 0. 0 citation(s).

[37] AM Turing. A quarterly review. -, 0. 0 citation(s).

[38] AM Turing. Ro gandy an early proof of normalization by am turing. -, 0. 2 citation(s).

[39] AM Turing. see turing. -, 0. 1 citation(s).

[40] AM Turing. The state of the art. -, 0. 3 citation(s).

[41] AM Turing. Turing’s treatise on enigma. -, 0. 5 citation(s).

[42] AM Turing. Universite paris 8 vincennes saint-denis licence m2i & info+ mineures departement de mathematiques et d’histoire des sciences m.-j. durand-richard des ... -, 0. 0 citation(s).

[43] AM Turing. with 1952. the chemical basis of morphogenesis. -, 0. 5 citation(s).

[44] AM Turing. Alan turing. - homosexualfamilies.viublogs.org, 1912. 0 citation(s).

[45] AM Turing. Handwritten essay: Nature of spirit. Photocopy available in www. turingarchive. org, item C/ ... -, 1932. 2 citation(s).

[46] AM Turing. On the gaussian error function. Unpublished Fellowship Dissertation, King’s College ... -, 1934. 6 citation(s).

[47] AM Turing. Proceedings of the London Mathematical Society -, 1936. 2 citation(s).

[48] AM Turing. 1937. on computable numbers, with an application to the entscheidungsproblem. Proceedings of the London Mathematical Society ... -, 1936. 12 citation(s).

[49] AM Turing. 7 ,'on computable numbers, with an application to the entscheidungsproblem'. The Undecidable, Raven, Ewlett -, 1936. 2 citation(s).
[50] AM Turing. On computable numbers proc. Lond. Math. Soc. 2nd Series -, 1936. 6 citation(s).
[51] AM Turing. On computable numbers with an application to the entscheidugsproblem. Proceedings of the Mathematical Society, sÃ©rie 2 - citeulike.org, 1936. 33 citation(s).
[52] AM Turing. Proccedings of the london mathematical society. -, 1936. 2 citation(s).
[53] AM Turing... The undecidable. - Cambridge University Press, 1936. 5 citation(s).
[54] AM Turing... with an application to the entscheidungsproblem. Proc. London Math. Soc -, 1936. 121 citation(s).
[55] AM Turing. Journal of Symbolic Logic -, 1937. 3 citation(s).
[56] AM Turing. The Journal of Symbolic Logic -, 1937. 2 citation(s).
[57] AM Turing. The $mathfrakp$-function in $lambda - k$-conversion. Journal of Symbolic Logic - projecteuclid.org, 1937. 0 citation(s).
[58] AM Turing. Computability and-definability. Journal of Symbolic Logic -, 1937. 42 citation(s).
[59] AM Turing. Computability and l-definability. Journal of Symbolic Logic - JSTOR, 1937. 99 citation(s).
[60] AM Turing. Computability and l-definability. JSL -, 1937. 2 citation(s).
[61] AM Turing. Correction to turing (1936). Proceedings of the London Mathematical Society (2) -, 1937. 2 citation(s).
[62] AM Turing. On computable numbers, with an application to the entscheidungsproblem. Proceedings of the London Mathematical ... - plms.oxfordjournals.org, 1937. 3937 citation(s).
[63] AM Turing. On computable numbers, with an application to the entscheidungsproblem',¡ i¿ proceedings of the london mathematical society(2) 42. A correction in -, 1937. 2 citation(s).
[64] AM Turing. On computable numbers, with an application to the entscheidungsproblem (paper read 12 november 1936). Proceedings of the London Mathematical Society -, 1937. 4 citation(s).
[65] AM Turing. The p-function in l-k-conversion. Journal of Symbolic Logic - JSTOR, 1937. 13 citation(s).
[66] AM Turing. The p functions in k conversion. J. Symbolic Logic -, 1937. 7 citation(s).
[67] AM Turing. Finite approximations to lie groups. Annals of Mathematics - JSTOR, 1938. 4 citation(s).
[68] AM Turing. Ox computable numbers, with an application to the entscheidungsproblem. J. of Math - l3d.cs.colorado.edu, 1938. 213 citation(s).
[69] AM Turing. Systems of logic based on ordinals: a dissertation. - Ph. D. dissertation, Cambridge ..., 1938. 1 citation(s).
[70] AM Turing. Systems of logic based on ordinals. Proceedings of the London Mathematical ... - plms.oxfordjournals.org, 1939. 350 citation(s).
[71] AM Turing. Systems of logic defined by ordinals. Procedings of the London Mathematical Society -, 1939. 8 citation(s).
[72] AM Turing. Mathematical theory of enigma machine. Public Record Office, London -, 1940. 3 citation(s).
[73] AM Turing. Proof that every typed formula has a normal form. Manuscript undated but probably -, 1941. 2 citation(s).
[74] AM Turing. The use of dots as brackets in church's system. Journal of Symbolic Logic - JSTOR, 1942. 2 citation(s).
[75] AM Turing. National Archives (London), box HW -, 1944. 2 citation(s).
[76] AM Turing. A method for the calculation of the zeta-function. Proceedings of the London Mathematical ... - plms.oxfordjournals.org, 1945. 16 citation(s).
[77] AM Turing. Proposal for development in the mathematical division of an automatic computing engine (ace)', reprinted in ince (1992). -, 1945. 2 citation(s).
[78] AM Turing. Proposed electronic calculator; reprinted in (copeland, 2005). A digital facsimile of the original typescript is available ... -, 1945. 2 citation(s).
[79] AM Turing. Proposed electronic calculator, copy of typescript available at www. turingarchive. org, item c/32. text published in various forms, eg in the collected ... DC Ince (North-Holland, 1992) -, 1946. 2 citation(s).
[80] AM Turing. Proposed electronic calculator, report for national physical laboratory, teddington. AM Turing's ACE Report of -, 1946. 2 citation(s).
[81] AM Turing. Proposed electronic calculator, report for national physical laboratory, teddington; published in am turing's ace report of 1946 and other papers, eds. ... - Cambridge, Mass.: MIT Press (1986), 1946. 2 citation(s).
[82] AM Turing. Lecture on the automatic computing engine; reprinted in (copeland, 2004). -, 1947. 2 citation(s).
[83] AM Turing. Lecture to the london mathematical society, 20 february 1947, typescript available at www. turingarchive. org, item b/1. text published in various forms, ... DC Ince (North-Holland, 1992) -, 1947. 2 citation(s).
[84] AM Turing. The state of the art. vortrag vor der londoner mathematical society am 20. februar 1947. Alan M. Turing, Intelligence Service. Schriften hrsg. von ... -, 1947. 2 citation(s).
[85] AM Turing. Intelligent machinery. mechanical intelligence. d. ince. - Amsterdam, North-Holland, 1948. 2 citation(s).
[86] AM Turing. Intelligent machinery-national physical laboratory report. b. meltzer b., d. michie, d.(eds) 1969, machine intelligence 5. - Edinburgh: Edinburgh University ..., 1948. 2 citation(s).
[87] AM Turing. Intelligent machinery, national physical laboratory report, typescript available at www. turingarchive. org, item c/11. text published in various forms, eg ... BJ Copeland (Oxford University Press, 2004) -, 1948. 2 citation(s).
[88] AM Turing. Intelligent machinery. npl report of the controller. - HMSO, 1948. 2 citation(s).
[89] AM Turing. Intelligent machinery. report for national physical laboratory. reprinted in ince, dc (editor). 1992. mechanical intelligence: Collected works of am turing. - Amsterdam: North Holland, 1948. 2 citation(s).
[90] AM Turing. Intelligent machinery', reprinted in ince (1992). -, 1948. 2 citation(s).
[91] AM Turing. Intelligent machinery. reprinted in ince, dc (editor). 1992. Mechanical Intelligence: Collected Works of AM Turing -, 1948. 4 citation(s).
[92] AM Turing. Practical forms of type theory. Journal of Symbolic Logic - JSTOR, 1948. 6 citation(s).
[93] AM Turing. Rounding-o errors in matrix processes. Quart. J. Mech. Appl. Math -, 1948. 10 citation(s).
[94] AM Turing. Rounding off-emfs in $matrdotsxp$ mcesses dagger quart. J. Mech. Appl. Math -, 1948. 0 citation(s).
[95] AM Turing. Rounding-off errors in matrix processes. The Quarterly Journal of Mechanics and Applied ... - Oxford Univ Press, 1948. 206 citation(s).
[96] AM Turing. Checking a large routine, report of a conference on high speed automatic calculating machines. Paper for the EDSAC Inaugural Conference -, 1949. 7 citation(s).
[97] AM Turing. Reprinted in Boden -, 1950. 2 citation(s).
[98] AM Turing. Aug s l doi. MIND - lcc.gatech.edu, 1950. 0 citation(s).
[99] AM Turing. Computer machinery and intelligence. Mind -, 1950. 46 citation(s).
[100] AM Turing. Computing machinery and intelligence', mind 59. -, 1950. 2 citation(s).
[101] AM Turing. Computing machinery and intelligence. mind lix (236): "460. bona fide field of study. he has cochaired the aaai fall 2005 symposium on machine ... IEEE Intelligent Systems -, 1950. 2 citation(s).
[102] AM Turing. Les ordinateurs et l'intelligence. Anderson, AR (1964) pp -, 1950. 6 citation(s).
[103] AM Turing. Macchine calcolatrici e intelligenza. Intelligenza meccanica - swif.uniba.it, 1950. 3 citation(s).
[104] AM Turing... Minds and machines. - Prentice-Hall Englewood Cliffs, NJ, 1950. 2 citation(s).
[105] AM Turing. Programmers. ... for Manchester Electronic Computer'. University of ... -, 1950. 5 citation(s).
[106] AM Turing. The word problem in semi-groups with cancellation. Annals of Mathematics - JSTOR, 1950. 33 citation(s).
[107] AM Turing. Can digital computers think?; reprinted in (copeland, 2004). -, 1951. 2 citation(s).
[108] AM Turing. Intelligent machinery, a heretical theory; reprinted in (copeland, 2004). -, 1951. 2 citation(s).
[109] AM Turing. Programmers' handbook for manchester electronic computer. University of Manchester Computing Laboratory -, 1951. 12 citation(s).
[110] AM Turing. Can automatic calculating machines be said to think?; reprinted in (copeland, 2004). -, 1952. 2 citation(s).

[111] AM Turing. The chemical bases of morphogenesis (reprinted in am turing' morphogenesis', north holland, 1992). -, 1952. 2 citation(s).
[112] AM Turing. A chemical basis for biological morphogenesis. Phil. Trans. Roy. Soc.(London), Ser. B -, 1952. 7 citation(s).
[113] AM Turing. The chemical basis of microphogenesis. Philos. Trans. R. Soc. B -, 1952. 3 citation(s).
[114] AM Turing. The chemical basis of morphogenesis. ... Transactions of the Royal Society of ... - rstb.royalsocietypublishing.org, 1952. 4551 citation(s).
[115] AM Turing. The chemical theory of 185. morphogenesis. Phil. Trans. Roy. Soc. B -, 1952. 7 citation(s).
[116] AM Turing. The chemical theory of morphogenesis. Phil. Trans. Roy. Soc -, 1952. 13 citation(s).
[117] AM Turing. Phil. trans. r. soc. B -, 1952. 2 citation(s).
[118] AM Turing. Philos. T rans. R. Soc. London -, 1952. 2 citation(s).
[119] AM Turing. Philos. trans. r. Soc. Ser. B -, 1952. 1 citation(s).
[120] AM Turing. Philosophical transactions of the royal society of london. series b. Biological Sciences -, 1952. 3 citation(s).
[121] AM Turing. The physical basis of morphogenesis. Phil. Trans. R. Soc -, 1952. 5 citation(s).
[122] AM Turing. Thechemical basis of moprhogenesis. Philosophical Transactions of the Royal Society of ... -, 1952. 5 citation(s).
[123] AM Turing. A theory of morphogenesis. Phil. Trans. B -, 1952. 12 citation(s).
[124] AM Turing. Chess; reprinted in (copeland, 2004). -, 1953. 2 citation(s).
[125] AM Turing. Digital computers applied to games. faster than thought. - Pitman Publishing, London, England ..., 1953. 5 citation(s).
[126] AM Turing. Faster than thought. Pitman, New York -, 1953. 4 citation(s).
[127] AM Turing. Review: Arthur w. burks, the logic of programming electronic digital computers. Journal of Symbolic Logic - projecteuclid.org, 1953. 0 citation(s).
[128] AM Turing. Some calculations of the riemann zeta-function. Proceedings of the London Mathematical ... - plms.oxfordjournals.org, 1953. 41 citation(s).
[129] AM Turing. Solvable and unsolvable problems. Science News - ens.fr, 1954. 39 citation(s).
[130] AM Turing. Can a machine think? in, newman, jr the world of mathematics. vol. iv. - New York: Simon and Schuster, Inc, 1956. 1 citation(s).
[131] AM Turing. Can a machine think? the world of mathematics. New York: Simon and Schuster -, 1956. 1 citation(s).
[132] AM TURING. Can a machine think? the world of mathematics. vol. 4, jr neuman, editor. - New York: Simon & Schuster, 1956. 3 citation(s).
[133] AM Turing. In' the world of mathematics'(jr newman, ed.), vol. iv. - Simon and Schuster, New York, 1956. 4 citation(s).
[134] AM TURING. Trees. US Patent 2,799,449 - Google Patents, 1957. 16 citation(s).
[135] AM TURING... In turing. - users.auth.gr, 1959. 2 citation(s).
[136] AM Turing. Intelligent machinery: A heretical view'. i¿ Alan M. Turing, Cambridge: Heffer & Sons -, 1959. 2 citation(s).
[137] AM Turing. Mind. Minds and machines. Englewood Cliffs, NJ: Prentice- ... -, 1964. 6 citation(s).
[138] AM Turing. Kann eine maschine denken. - Kursbuch, 1967. 45 citation(s).
[139] AM Turing. Intelligent machinery, report, national physics laboratory, 1948. reprinted in: B. meltzer and d. michie, eds., machine intelligence 5. - Edinburgh University Press, ..., 1969. 3 citation(s).
[140] AM Turing... Am turing's original proposal for the development of an electronic computer: Reprinted with a foreword by dw davies. - National Physical Laboratory, ..., 1972. 1 citation(s).
[141] AM Turing. Maszyny liczace a inteligencja, taum. - ... i malenie, red. E. Feigenbaum, J. ..., 1972. 3 citation(s).
[142] AM Turing. A quarterly review of psychology and philosophy. Pattern recognition: introduction and ... - Dowden, Hutchinson & Ross Inc., 1973. 0 citation(s).
[143] AM TURING. Puede pensar una maquina? trad. cast. de m. garrido y a. anton. Cuadernos Teorema, Valencia -, 1974. 2 citation(s).
[144] AM Turing. Dictionary of scientific biography xiii. -, 1976. 0 citation(s).
[145] AM Turing. Artificial intelligence: Usfssg computers to think about thinking. part 1. representing knowledge. - Citeseer, 1983. 0 citation(s).
[146] AM TURING. The automatic computing machine: Papers by alan turing and michael woodger. - MIT Press, Cambridge, MA, 1985. 2 citation(s).
[147] AM Turing... The automatic computing engine: Papers by alan turing and michael woodger. - mitpress.mit.edu, 1986. 0 citation(s).
[148] AM Turing. Proposal for development in the mathematics division of an automatic computing engine (ace). Carpenter, BE, Doran, RW (eds) -, 1986. 46 citation(s).
[149] AM Turing. Jones, jp, and yv majjjasevic 1984 register machine proof of the theorem on exponential diophamine-representation of enumerable sets. j. symb. log. 49 (1984) ... Information, randomness & incompleteness: papers ... - books.google.com, 1987. 0 citation(s).
[150] AM Turing. Rechenmaschinen und intelligenz. Alan Turing: Intelligence Service (S. 182). Berlin: ... -, 1987. 8 citation(s).
[151] AM Turing. Rounding-off errors in matrix processes, quart. J. Mech -, 1987. 10 citation(s).
[152] AM Turing. Can a machine think? The World of mathematics: a small library of the ... - Microsoft Pr, 1988. 104 citation(s).
[153] AM Turing. Local programming methods and conventions. The early British computer conferences - portal.acm.org, 1989. 1 citation(s).
[154] AM Turing. The chemical basis of morphogenesis. 1953. Bulletin of mathematical biology - ncbi.nlm.nih.gov, 1990. 28 citation(s).
[155] AM Turing. The chemical basis of morphogenesis, reprinted from philosophical transactions of the royal society (part b), 237, 37-72 (1953). Bull. Math. Biol -, 1990. 2 citation(s).
[156] AM Turing. 2001. Collected works of aM Turing -, 1992. 1 citation(s).
[157] AM Turing. Collected works of alan turing, morphogenesis. - by PT Saunders. Amsterdam: ..., 1992. 1 citation(s).
[158] AM Turing. The collected works of am turing: Mechanical intelligence,(dc ince, ed.). - North-Holland, 1992. 3 citation(s).
[159] AM Turing. Collected works, vol. 3: Morphogenesis (pt saunders, editor). - Elsevier, Amsterdam, New York, ..., 1992. 3 citation(s).
[160] AM Turing... A diffusion reaction theory of morphogenesis in plants. Collected Works of AM Turing: Morphogenesis, PT ... -, 1992. 4 citation(s).
[161] AM Turing. Intelligent machinery (written in 1947.). Collected Works of AM Turing: Mechanical Intelligence. ... -, 1992. 2 citation(s).
[162] AM Turing. Intelligent machines. Ince, DC (Ed.) -, 1992. 5 citation(s).
[163] AM Turing. Lecture to the london mathematical society. The Collected Works of AM Turing, volume Mechanical ... -, 1992. 5 citation(s).
[164] AM Turing... Mechanical intelligence. - cdsweb.cern.ch, 1992. 25 citation(s).
[165] AM Turing... Morphogenesis. - North Holland, 1992. 5 citation(s).
[166] AM Turing. Morphogenesis. collected works of am turing, ed. pt saunders. - Amsterdam: North-Holland, 1992. 2 citation(s).
[167] AM Turing... Intelligenza meccanica. - Bollati Boringhieri, 1994. 4 citation(s).
[168] AM Turing. Lecture to the london mathematical society on 20 february 1947. MD COMPUTING - SPRINGER VERLAG KG, 1995. 64 citation(s).
[169] AM Turing. Theorie des nombres calculables, suivi d'une application au probleme de la decision. La machine de Turing -, 1995. 4 citation(s).
[170] AM Turing. I calcolatori digitali possono pensare? Sistemi intelligenti - security.mulino.it, 1998. 0 citation(s).
[171] AM Turing. Si pui dire che i calcolatori automatici pensano? Sistemi intelligenti - mulino.it, 1998. 0 citation(s).
[172] AM Turing. Collected works: Mathematical logic amsterdam etc. - North-Holland, 2001. 7 citation(s).
[173] AM Turing. Collected works: Mathematical logic (ro gandy and cem yates, editors). - Elsevier, Amsterdam, New York, ..., 2001. 10 citation(s).
[174] AM Turing. Visit to national cash register corporation of dayton, ohio. Cryptologia - Taylor & Francis Francis, 2001. 0 citation(s).
[175] AM Turing. Alan m. turing's critique of running short cribs on the us navy bombe. Cryptologia - Taylor & Francis, 2003. 0 citation(s).
[176] AM Turing. Can digital computers think? The Turing test: verbal behavior as the hallmark of ... - books.google.com, 2004. 27 citation(s).
[177] AM Turing. Computing machinery and intelligence. 1950. The essential Turing: seminal writings in computing ... - books.google.com, 2004. 13 citation(s).
[178] AM Turing... The essential turing. - Clarendon Press, 2004. 2 citation(s).

[179] AM Turing. Intelligent machinery, a heretical theory. The Turing test: verbal behavior as the hallmark of ... - books.google.com, 2004. 264 citation(s).
[180] AM Turing. Lecture on the a utomatic computing e ngine, 1947. BJ Dopeland(E d.), The E ssential Turing, O UP -, 2004. 1 citation(s).
[181] AM Turing. Retrieved july 19, 2004. -, 2004. 2 citation(s).
[182] AM Turing. The undecidable: Basic papers on undecidable propositions, unsolvable problems and computable functions. - Dover Mineola, NY, 2004. 4 citation(s).
[183] AM Turing. 20. proposed electronic calculator (1945). Alan Turing 39; s Automatic Computing Engine - ingentaconnect.com, 2005. 0 citation(s).
[184] AM Turing. 21. notes on memory (1945). Alan Turing 39; s Automatic Computing Engine - ingentaconnect.com, 2005. 0 citation(s).
[185] AM Turing... 22. the turingwilkinson lecture series (19467). Alan Turing 39; s Automatic ... - ingentaconnect.com, 2005. 0 citation(s).
[186] AM Turing. Biological sequences and the exact string matching problem. Introduction to Computational Biology - Springer, 2006. 0 citation(s).
[187] AM Turing. Fernando j. elizondo garza. CIENCIA UANL - redalyc.uaemex.mx, 2008. 0 citation(s).
[188] AM Turing. Computing machinery and intelligence. Parsing the Turing Test - Springer, 2009. 4221 citation(s).
[189] AM Turing. Equivalence of left and right almost periodicity. Journal of the London Mathematical Society - jlms.oxfordjournals.org, 2009. 2 citation(s).
[190] AM Turing. A study of logic and programming via turing machines. ... : classroom projects, history modules, and articles - books.google.com, 2009. 0 citation(s).
[191] AM Turing, MA Bates, and BV Bowden... Digital computers applied to games. Faster than thought -, 1953. 101 citation(s).
[192] AM Turing, BA Bernstein, and R Peter... Logic based on inclusion and abstraction wv quine; 145-152. Journal of Symbolic ... - projecteuclid.org, 2010. 0 citation(s).
[193] AM Turing, R Braithwaite, and G Jefferson... Can automatic calculating machines be said to think? Copeland (1999) -, 1952. 17 citation(s).
[194] AM Turing and JL Britton... Pure mathematics. - North Holland, 1992. 1 citation(s).
[195] AM Turing and BE Carpenter... Am turing's ace report of 1946 and other papers. - MIT Press, 1986. 6 citation(s).
[196] AM Turing and BJ Copel... Book review the essential turing reviewed by andrew hodges the essential turing. -, 2008. 0 citation(s).
[197] AM Turing and B Dotzler... Intelligence service: Schriften. - Brinkmann & Bose, 1987. 27 citation(s).
[198] AM Turing and EA Feigenbaum... Computers and thought. Computing Machinery and Intelligence, EA ... -, 1963. 6 citation(s).
[199] AM Turing and RO Gandy... Mathematical logic. - books.google.com, 2001. 2 citation(s).
[200] AM Turing, M Garrido, and A Anton... Puede pensar una maquina? - ... de Logica y Filosofia de la Ciencia, 1974. 12 citation(s).
[201] AM Turing, JY Girard, and J Basch... La machine de turing. - dil.univ-mrs.fr, 1995. 26 citation(s).
[202] AM Turing and DR Hofstadter... The mind's. - Harvester Press, 1981. 3 citation(s).
[203] AM Turing, D Ince, and JL Britton... Collected works of am turing. - North-Holland Amsterdam, 1992. 17 citation(s).
[204] AM Turing and A Lerner... Aaai 1991 spring symposium series reports. 12 (4): Winter 1991, 31-37 aaai 1993 fall symposium reports. 15 (1): Spring 1994, 14-17 aaai 1994 spring ... Intelligence - aaai.org, 1987. 0 citation(s).
[205] AM Turing and P Millican... Machines and thought: Connectionism, concepts, and folk psychology. - Clarendon Press, 1996. 0 citation(s).
[206] AM Turing and P Millican... Machines and thought: Machines and thought. - Clarendon Press, 1996. 0 citation(s).
[207] AM Turing and PJR Millican... The legacy of alan turing. -, 0. 3 citation(s).
[208] AM Turing and PJR Millican... The legacy of alan turing: Connectionism, concepts, and folk psychology. - Clarendon Press, 1996. 0 citation(s).
[209] AM Turing, J Neumann, and SA Anovskaa... Mozet li masina myslit'? - Gosudarstvennoe Izdatel'stvo Fiziko- ..., 1960. 2 citation(s).
[210] AM Turing and H Putnam... Mentes y maquinas. - Tecnos, 1985. 3 citation(s).
[211] AM Turing, C Works, SB Cooper, and YL Ershov... Computational complexity theory. -, 0. 0 citation(s).
[212] FRS AM TURING. The chemical basis of morphogenesis. Sciences - cecm.usp.br, 1952. 0 citation(s).

This script is sent with the consent of the B.B.C. GB.

AMT/B/6/2

NOT CHECKED IN TALKS DEPARTMENT WITH 'AS BROADCAST' SCRIPT

Producer: T. S. Gregory.

CAN AUTOMATIC CALCULATING MACHINES BE SAID TO THINK?

Discussion between:-

M. H. A. Newman, F.R.S.,
Professor of Mathematics in
the University of Manchester.

A. M. Turing, F.R.S.,
~~Lecturer in Mathematics~~ Reader in
the University of Manchester.

Sir Geoffrey Jefferson, F.R.S.,
Professor of Neuro-Surgery in
the University of Manchester.

R. B. Braithwaite, Fellow of
King's College, Cambridge.

Recording: Thursday, January 10, 1952.
Disc No.: SLO.1578 Duration:

Transmission: Monday, January 14, 1952. 9.35 - 10.20 THIRD PROGRAMME

Repeat: Wednesday, January 23, 1952. 6.20 - 7.5 THIRD PROGRAMME

BRAITHWAITE: We're here today to discuss whether calculating machines can be said to think in any proper sense of the word. Thinking is ordinarily regarded as so much a speciality of man, and perhaps of other higher animals, that the question may seem too absurd to be discussed. But, of course, it all depends on what is to be included in thinking. The word is used to cover a multitude of different activities. What would you, Jefferson, as a physiologist, say were the most important elements involved in thinking?

JEFFERSON: I don't think that we need waste much time on definition of thinking since it will be hard to get beyond phrases in common usage, such as having ideas in the mind, cogitating, meditating, deliberating, solving problems or imagining. Philologists say that the word "Man" is derived from a Sanskrit word that means "to think," probably in the sense of judging between one idea and another. I agree that we could no longer use the word thinking in a sense that restricted it to man. No one would deny that many animals think, though in a very limited way. They lack insight. For example, a dog learns that it is wrong to get on cushions or chairs with muddy paws, but he only learns it as a venture that doesn't pay. He has no conception of the real reason, that he damages fabrics by doing that.

The average person would perhaps be content to define thinking in very general terms such as revolving ideas in the mind, of having notions in one's head, of having one's mind occupied by a problem and so on. But it is only right to add that our minds are occupied much of the time with trivialities. One might say in the end that thinking was the general result of having a sufficiently complex nervous system. Very simple ones do not provide the creature with any problems that are not answered by simple reflex mechanisms. Thinking then becomes all the things that go on in one's brain, things that often end in an action but don't necessarily do so. I should say that it was the sum total of what the brain of man or animal does.

Turing, what do you think about it? Have you a mechanical definition?

<u>TURING</u>: I don't want to give a definition of thinking, but if I had to I should probably be unable to say anything more about it than that it was a sort of buzzing that went on inside my head. But I don't really see that we need to agree on a definition at all. The important thing is to try to draw a

line between the properties of a brain, or of a man, that we want to discuss, and those that we don't. To take an extreme case, we are not interested in the fact that the brain has the consistency of cold porridge. We don't want to say "This machine's quite hard, so it isn't a brain, and so it can't think." I would like to suggest a particular kind of test that one might apply to a machine. You might call it a test to see whether the machine thinks, but it would be better to avoid begging the question, and say that the machines that pass are (let's say) Grade A. machines. The idea of the test is that the machine has to try and pretend to be a man, by answering questions put to it, and it will only pass if the pretence is reasonably convincing. A considerable proportion of a jury, who should not be expert about machines, must be taken in by the pretence. They aren't allowed to see the machine itself - that would make it too easy. So the machine is kept in a far away room and the jury are allowed to ask it questions, which are transmitted through to it: it sends back a typewritten answer.

BRAITHWAITE: Would the questions have to be sums, or could I ask it what it had had for breakfast?

TURING: Oh yes, anything. And the questions don't really have to be questions, any more than questions in a law court are really questions. You know the sort of thing. "I put it to you that you are only pretending to be a man" would be quite in order. Likewise the machine would be permitted all sorts of tricks so as to appear more man-like, such as waiting a bit before giving the answer, or making spelling mistakes, but it can't make smudges on the paper, any more than one can send smudges by telegraph. We had better suppose that each jury has to judge quite a number of times, and that sometimes they really are dealing with a man and not a machine. That will prevent them saying "It must be a machine" every time without proper consideration.

Well, that's my test. Of course I am not saying at present either that machines really could pass the test, or that they couldn't. My suggestion is just that this is the question we should discuss.

AMT/B/6/7

It's not the same as "Do machines think," but it seems near enough for our present purpose, and raises much the same difficulties.

NEWMAN: I should like to be there when your match between a man and a machine takes place, and perhaps to try my hand at making up some of the questions. But that will be a long time from now, if the machine is to stand any chance with no questions barred?

TURING: Oh yes, at least 100 years, I should say.

JEFFERSON: Newman, how well would existing machines stand up to this test? What kind of things can they do now?

NEWMAN: Of course their strongest line is mathematical computing, which they were designed to do, but they would also do well at some questions that don't look numerical, but can easily be made so, like solving a chess problem or looking you up a train in the time-table.

BRAITHWAITE: Could they do that?

NEWMAN: Yes. Both these jobs can be done by trying all the possibilities, one after another. The whole of the information in an ordinary time-table would have to be written in as part of the programme, and the simplest possible routine would be one that found the trains from London to Manchester by testing every train in the time-table to see if it calls at both places, and printing out those that do. Of course this is a dull plodding method, and you could improve on it by using a more complicated routine, but if I have understood Turing's test properly, you are not allowed to go behind the scenes and criticise the method, but must abide by the scoring on correct answers, found reasonably quickly.

JEFFERSON: Yes, but all the same a man who has to look up trains frequently gets better at it, as he learns his way about the time-table. Suppose I give a machine the same problem again, can it learn to do better without going through the whole rigmarole of trying everything over every time? I'd like to have your answer to that because it's such an important point. Can machines learn to do better with practice?

NEWMAN: Yes, it could. Perhaps the chess problem provides a better illustration of this. First I should mention that all the information required in any job - the numbers, times of trains, positions of pieces, or whatever it is, and also the instructions saying what is to be done with them - all this material is stored in the same way. (In the Manchester machine it is stored as a pattern on something resembling a television screen.) As the work goes on the pattern is changed. Usually it is the part of the pattern that contains the data that changes, while the instructions stay fixed. But it is just as simple to arrange that the instructions themselves shall be changed now and then. Well, now a programme could be composed that would cause the machine to do this: a 2-move chess problem is recorded into the machine in some suitable coding, and whenever the machine is started, a white move is chosen at random (there is a device for making random choices in our machine). All the consequences of this move are now analysed, and if it does not lead to forced mate in two moves, the machine prints, say, "P-Q3, wrong move," and stops. But the analysis shows that when the right move is chosen the machine not only prints, say, "B-Q5, solution," but it changes the instruction calling for a random choice to one that says "Try B-Q5." The result is that whenever the machine is started again it will immediately print out the right solution - and this without the man who made up the routine knowing beforehand what it was. Such a routine could certainly be made now, and I think this can fairly be called learning.

JEFFERSON: Yes, I suppose it is. Human beings learn by repeating the same exercises until they have perfected them. Of course it goes further and at the same time - we learn generally to shift the knowledge gained about one thing to another set of problems, seeing relevances and relationships. Learning means remembering. How long can a machine store information for?

NEWMAN: Oh, at least as long as a man's lifetime, if it is refreshed occasionally.

JEFFERSON: Another difference would be that in the learning process there is much more frequent intervention by teachers, parental or otherwise, guiding the arts of learning. You mathematicians put the programme once into the machine and leave it to it. You wouldn't get any distance at all with human beings if that is what you did. In fact, the only time you do that in the learning period is at examinations.

TURING: It's quite true that when a child is being taught, his parents and teachers are repeatedly intervening to stop him doing this or encourage him to do that. But this will not be any the less so when one is trying to teach a machine. I have made some experiments in teaching a machine to do some simple operation, and a very great deal of such intervention was needed before I could get any results at all. In other words the machine learnt so slowly that it needed a great deal of teaching.

JEFFERSON: But who was learning, you or the machine?

TURING: Well, I suppose we both were. One will have to find out how to make machines that will learn more quickly if there is to be any real success. One hopes too that there will be a sort of snowball effect. The more things the machine has learnt the easier it ought to be for it to learn others. In learning to do any particular thing it will probably also be learning to learn more efficiently. I am inclined to believe that when one has taught it to do certain things one will find that some other things which one had planned to teach it are happening without any special teaching being required. This certainly happens with an intelligent human mind, and if it doesn't happen when one is teaching a machine there is something lacking in the machine. What do you think about learning possibilities, Braithwaite?

BRAITHWAITE: No one has mentioned what seems to me the great difficulty about learning, since we've only discussed learning to solve a particular problem. But the most important part of human learning is learning from experience - Not learning from one particular kind of experience, but being able to learn from experience in general. A machine can easily be constructed with a feed-back device so that the programming of the machine is controlled by the relation of its output to some feature in its external environment - so that the working of the machine in relation to the environment is self-corrective. But this requires that it should be some particular feature of the environment to which the machine has to adjust itself. The peculiarity of men and animals is that they have the power of adjusting themselves to almost all the features. The feature to which adjustment is made on a particular occasion is the one the man is attending to and he attends to what he is <u>interested in</u>. His interests are determined, by and large, by his appetites, desires, drives, instincs - all the things that together make up his "springs of action." If we want to conruct a machine which will vary its attention to things in its environment so that it will sometime adjust itself to one and sometimes to another, it would seem to be necessary to equip the machine with something corresponding to a set of appetites. If the machine is built to be treated only as a domestic pet, and is spoon-fed with particular problems, it will not be able to learn in the varying way in which human beings learn. This arises from the necessity of adapting behaviour suitably to environment if human appetites are to be satisfied.

<u>JEFFERSON</u>: Turing, you spoke with great confidence about what you are going to be able to do. You make it sound as if it would be fairly easy to modify construction so that the machine reacted more like a man. But I recollect that from the time of Descartes and Borelli on people have said that it would be only a matter of a few years, perhaps 3 or 4 or maybe 50, and a replica of man would have been artificially created. We shall be wrong, I am sure, if we give the impression that these things would be easy to do.

<u>NEWMAN</u>: I agree that we are getting rather far away from computing machines as they exist at present. These machines have rather restricted appetites, and they

can't blush when they're embarrassed, but it's quite hard enough, and I think a very interesting problem, to discover how near these actually existing machines can get to thinking. Even if we stick to the reasoning side of thinking, it is a long way from solving chess problems to the invention of new mathematical concepts or making a generalisation that takes in ideas that were current before, but had never been brought together as instances of a single general notion.

BRAITHWAITE: For example?

NEWMAN: The different kinds of number. There are the integers, 0, 1, -2, and so on; there are the real numbers used in comparing lengths, for example the circumference of a circle and its diameter; and the complex numbers involving $\sqrt{-1}$; and so on. It is not at all obvious that these are instances of one thing, "number." The Greek mathematicians used entirely different words for the integers and the real numbers, and had no single idea to cover both. It is really only recently that the general notion of kinds of number, had been abstracted from these instances and accurately defined. To make this sort of generalisation you need to have the power of recognising similarities, seeing analogies between things that had not been put together before. It is not just a matter of testing things for a specified property and classifying them accordingly. The concept itself has to be framed, something has to be created, say the idea of a number-field. Can we even guess at the way a machine could make such an invention from a programme composed by a man who had not the concept in his own mind?

TURING: It seems to me, Newman, that what you said about "trying out possibilities" as a method applies to quite an extent, even when a machine is required to do something as advanced as finding a useful new concept. I wouldn't like to have to define the meaning of the word "concept," nor to give rules for rating their usefulness, but whatever they are they've got outward and visible forms, which are words and combinations of words. A machine could make up such combinations of words more or less at random, and then give them marks for various merits.

NEWMAN: Wouldn't that take a prohibitively long time?

TURING: It would certainly be shockingly slow, but it could start on easy things, such as lumping together rain, hail, snow and sleet, under the word 'precipitation.' Perhaps it might do more difficult things later on if it was learning all the time how to improve its methods.

BRAITHWAITE: I don't think there's much difficulty about seeing analogies that can be formally analysed and explicitly stated. It is then only a question of designing the machine so that it can recognise similarities of mathematical structure. The difficulty arises if the analogy is a vague one about which little more can be said than that one has a feeling that there is some sort of similarity between two cases but one hasn't any idea as to the respect in which the two cases are similar. A machine can't recognise similarities when there is nothing in its programme to say what are the similarities it is expected to recognise.

TURING: I think you could make a machine spot an analogy, in fact it's quite a good instance of how a machine could be made to do some of those things that one usually regards as essentially a human monopoly. Suppose that someone was trying to explain the double negative to me, for instance, that when something isn't green it must be green, and he couldn't quite get it across. He might say, 'Well, it's like crossing the road. You cross it, and then you cross it again, and you're back where you started.' This remark might just clinch it. This is one of the things one would like to work with machines, and I think it would be likely to happen with them. I imagine that the way analogy works in our brains is something like this. When two or more sets of ideas have the same pattern of logical connections, the brain may very likely economise parts by using some of them twice over, to remember the logical connections both in the one case and in the other. One must suppose that some part of my brain was used twice over in this way, once for the idea of double negation and once for crossing the road, there and back. I am really supposed to know about both these things but can't get what it is the man is driving at, so long as he is talking about all these dreary nots and not-nots. Somehow it doesn't get through to the right part of the brain. But as soon as he says his piece about crossing the road it gets through to the right part, but by a different route. If there is some such purely mechanical explanation of how this argument by analogy goes on in the brain, one could make a digital computer do the same.

<u>JEFFERSON</u>: Well, there isn't a mechanical explanation in terms of cells and connecting fibres in the brain.

<u>BRAITHWAITE</u>: But could a machine really do this? How would it do it?

<u>TURING</u>: I've certainly left a great deal to the imagination. If I had given a longer explanation I might have made it seem more certain that what I was describing was feasible, but you would probably exclaim impatiently, 'Well, yes, I see that a machine could do all that, but I wouldn't call it thinking.' As soon as one can see the cause

and effect working themselves out in the brain, one regards it as not being thinking, but a sort of unimaginative donkey-work. From this point of view one might be tempted to define thinking as consisting of "those mental processes that we don't understand." If this is right then to make a thinking machine is to make one which does interesting things without our really understanding quite how it is done.

JEFFERSON: If you mean that we don't know the wiring in men, as it were, that is quite true.

TURING: No, that isn't at all what I mean. We know the wiring of our machine, but it already happens there in a limited sort of way. Sometimes a computing machine does do something rather weird that we hadn't expected. In principle one could have predicted it, but in practice it's usually too much trouble. Obviously if one were to predict everything a computer was going to do one might just as well do without it.

NEWMAN: It is quite true that people are disappointed when they discover what the big computing machines actually do which is just to add and multiply, and use the results to decide what further additions and multiplications to do. '<u>That's</u> not thinking', is the natural comment, but this is rather begging the question. If you go into one of the ancient churches in Ravenna you see some most beautiful pictures round the walls, but if you peer at them through binoculars you might say, 'Why, they aren't really pictures at all, but just a lot of little coloured stones with cement in between.' The machines processes are mosaics of very simple standard parts, but the designs can be of great complexity, and it is not obvious where the limit is to the patterns of thought they could imitate.

<u>BRAITHWAITE</u>: But how many stones are there in your mosaic? Jefferson, is there a sufficient multiplicity of the cells in the brain for them to behave like a computing machine?

<u>JEFFERSON</u>: Yes, there are thousands, tens of thousands more cells in the brain than there are in a computing machine, because the present machine contains - how many did you say?

<u>TURING</u>: Half a million digits. I think we can assume that is the equivalent of half a million nerve cells.

<u>BRAITHWAITE</u>: If the brain works like a computing machine then the present computing machine cannot do all the things the brain does. Agreed; but if a computing machine were made that could do all the things the brain does, wouldn't it require more digits than there is room for in the brain?

<u>JEFFERSON</u>: Well, I don't know. Suppose that it is right to equate digits in a machine with nerve cells in a brain. There are various estimates, somewhere between ten thousand million and fifteen thousand million cells are supposed to be there. Nobody knows for certain, you see. It is a colossal number. You would need 20,000 or more of your machines to equate digits with nerve cells. But it is not, surely, just a question of size. There would be too much logic in your huge machine. It wouldn't be really like a human output of thought. To make it more like, a lot of the machine parts would have to be designed quite differently to give greater flexibility and more diverse possibilities of use. It's a very tall order indeed.

TURING: It really is the size that matters in this case. It is the amount of information that can be stored up. If you think of something very complicated that you want one of these machines to do, you may find the particular machine you have got won't do, but if any machine can do it at all, then it can be done by your first computer, simply increased in its storage capacity.

JEFFERSON: If we are really to get near to anything that can be truly called 'thinking' the effects of external stimuli cannot be missed out; the intervention of all sorts of extraneous factors, like the worries of having to make one's living, or pay one's taxes, or get food that one likes. These are not in any sense minor factors, they are very important indeed, and worries concerned with them may greatly interfere with good thinking, especially with creative thinking. You see a machine has not environment, and man is in constant relation to his environment, which, as it were punches him whilst he punches back. There is a vast background of memories in a man's brain that each new idea or experience has to fit in with. I wonder if you could tell me how far a calculating machine meets that situation. Most people agree that man's first reaction to a new idea (such as the one we are discussing today) is one of rejection, often immediate and horrified denial of it. I don't see how a machine could as it were say "now Professor Newman or Mr. Turing, I don't like this programme at all that you've just put it into me, in fact I'm not going to have anything to do with it!"

NEWMAN: One difficulty about answering that is one that Turing has already mentioned. If someone says, "Could a machine do this, e.g. could it say 'I don't like the programme you have just put into me'," and a programme for doing that very thing is duly produced, it is apt to have an artificial and ad hoc air, and appear to be more of a trick than a serious answer to the question. It is like those passages in the Bible, which worried me as a small boy, that say that such and such was done "that the prophecy might be fulfilled which says" so and so. This always seemed to me a most unfair way of making sure that the prophecy came true. If I answer your question, Jefferson, by making a routine which simply caused the machine to say just the words "Newman and Turing, I don't like your programme," you would certainly feel this was a rather childish trick, and not the answer to what you really wanted to know. But yet it's hard to pin down what you do want.

JEFFERSON: I want the machine to reject the problem because it offends it in some way. That leads me to enquire what the ingredients are of ideas that we reject because we instinctively don't care for them. I don't know why I like some pictures and some music and am bored by other sorts. But I'm not going to carry that line on because we are all different, our dislikes are based on our personal histories and probably too on small differences of construction in all of us, I mean by heredity. Your machines have no genes, no pedigrees. Mendelian inheritance means nothing to wireless valves. But I don't want to score debating points! We ought to make it clear that not even Turing thinks that all that he has to do is to put a skin on the machine and that it is alive! We've been trying for a more limited objective whether the sort of thing that machines do can be considered as thinking. But is not your machine more certain than any human being of getting its problem right at once, and infallibly?

NEWMAN: Oh !

TURING: Computing machines aren't really infallible at all. Making up checks on their accuracy is quite an important part of the art of using them. Besides making mistakes they sometimes haven't done quite the calculation one had expected, and one gets something that might be called a "misunderstanding."

JEFFERSON: At any rate, they are not influenced by the emotions. You have only to upset a person enough and he becomes confused, he can't think of the answers and may make a fool of himself. It is high emotional content of mental processes in the human being that makes him quite different from a machine. It seems to me to come from the great complexity of his nervous system with its 10^{10} cells and also from his endocrine system which imports all sorts of emotions and instincts, such as those to do with sex. Man is essentially a chemical machine, he is much effected by hunger and fatigue, by being "out of sorts" as we say, also by innate judgments, and by sexual urges. This chemical side is tremendously important, not the least so because the brain does exercise a remote control over the most important chemical processes that go on in our bodies. Your machines don't have to bother with that, with being tired or cold or happy or satisfied. They show no delight at having done something never done before. No, they are "mentally" simple things. I mean that however complicated their structure is (and I know it is very complicated) compared with man they are very simple and perform their tasks with an absence of distracting thoughts which is quite inhuman.

BRAITHWAITE: I'm not sure that I agree. I believe that it will be necessary to provide the machine with something corresponding to appetites, or other "springs of action," in order that it will pay enough attention to relevant features in its environment to be able to learn from experience. Many psychologists have held that the emotions in men are by-products of their appetites and that they serve a biological function in calling higher levels of mental activity into play when the lower levels are incapable of coping with an external situation. For example, one does not feel afraid when there is no danger, or a danger which can be avoided more or less automatically: fear is a sympton showing that the danger has to be met by conscious thought. Perhaps it will be impossible to build a machine capable of learning in general from experience without incorporating in it an emotional apparatus, the function of which will be to switch over to a different part of the machine when the external environment differs too much from what would satisfy the machines appetites by more than a certain amount. I don't want to suggest that it will be necessary for the machine to be able to throw a fit of tantrums. But in humans tantrums frequently fulfil a definite function - that of escaping from responsibility: and to protect a machine against a too hostile environment it may be essential to allow it, as it were, to go to bed with a neurosis, or psychogenic illness - just as, in a simpler way, it is provided with a fuse to blow, if the electric power working it threatens its continued existence.

TURING: Well, I don't envisage teaching the machine to throw temperamental scenes. I think some such effects are likely to occur as a sort of by-product of genuine teaching, and that one will be more interested in curbing such displays than in encouraging them. Such effects would probably be distinctly different from the corresponding human ones, but recognisable as variations on them. This means that if the machine was being put through one of my imitation tests, it would have to do quite a bit of acting, but if one was comparing it with a man in a less strict sort of way the resemblance might be quite impressive.

NEWMAN: I still feel that too much of our argument is about what hypothetical future machines will do. It is all very well to say that a machine could easily be made to do this or that, but, to take only one practical point, what about the time it would take to do it? It would only take an hour or two to make up a routine to make our Manchester machine analyse all possible variations of the game of chess right out, and find the best move that way - if you didn't mind its taking thousands of millions of years to run through the routine. Solving a problem on the machine doesn't mean finding a way to do it between now and eternity, but within a reasonable time. This is not just a technical detail that will be taken care of by future improvements. It's most unlikely that the engineers can ever give us a factor of more than a thousand or two times our present speeds. To assume that runs that would take thousands of millions of years on our present machines will be done in a flash on machines of the future, is to move into the realms of science fiction.

TURING: To my mind this time factor is the one question which will involve all the real technical programming difficulty. If one didn't know already that these things can be done by brains within a reasonable time one might think it hopeless to try with a machine. The fact that a brain can do it seems to suggest that the difficulties may not really be so bad as they now seem.

BRAITHWAITE: I agree that we ought not to extend our discussion to cover whether calculating machines could be made which would do everything that a man can do. The point is, surely, whether they can do all that it is proper to call thinking. Appreciation of a picture contains elements of thinking, but it also contains elements of feeling; and we're not concerned with whether a machine can be made that will feel. Similarly with moral questions: we're only concerned with them so far as they are also intellectual ones. We haven't got to give the machine a sense of duty or anything corresponding to a will: still less need it be given temptations which it would then have to have an apparatus for resisting. All that it has got to do in order to think is to be able to solve, or to make a good attempt at solving, all the intellectual problems with which it might be confronted by the environment in which it finds itself. This environment, of course, must include Turing asking it awkward questions as well as natural events such as being rained upon, or being shaken up by an earthquake.

NEWMAN: But I thought it was you who said that a machine wouldn't be able to learn to adjust to its environment if it hadn't been provided with a set of appetites and all that went with them?

<u>BRAITHWAITE</u>: Yes, certainly. But the problems raised by a machine having appetites are not properly our concern today. It may be the case that it wouldn't be able to learn from experience without them; but we're only required to consider whether it would be able to learn at all - since I agree that being able to learn is an essential part of thinking. So oughtn't we to get back to something centred on thinking? Can a machine make up new concepts, for example?

<u>NEWMAN</u>: There are really two questions that can be asked about machines and thinking, first, what do we require before we agree that the machine does <u>everything</u> that we call thinking? This is really what we have been talking about for most of the time; but there is also another interesting and important question: Where does the doubtful territory begin? What is the <u>nearest</u> thing to straight computing that the present machines perhaps can't do?

<u>BRAITHWAITE</u>: And what would your own answer be?

<u>NEWMAN</u>: I think perhaps to solve mathematical problems

for which no method is known, in the way that men do; to find new methods. This is a much more modest aim than inventing new mathematical concepts. What happens when you try to solve a new problem in the ordinary way is that you think about it for a few seconds, or a few years, trying out all the analogies you can think of with problems that have been solved, and then you have an idea. You try it out in detail. If it is no good you must wait for another idea. This is a little like the chess-problem routine, where one move after another is tried, but with one very important difference, that if I am even a moderately good mathematician the ideas that I get are not just random ones, but are pre-selected so that there is an appreciable chance that after a few trials one of them will be successful. Henry Moore says about the studies he does for his sculpture, "When the work is more than an exercise, inexplicable jumps occur. This is where the imagination comes in." If a machine could really be got to imitate this sudden pounce on an idea, I believe that everyone would agree that it had begun to think, even though it didn't have appetites or worry about the income tax. And suppose that we also stuck to what we knew about the physiology of human thinking, how much would that amount to, Jefferson?

JEFFERSON: We know a great deal about the end-product, thinking itself. Are not the contents of our libraries and museums the total up to date? Experimental psychology has taught us a lot about the way that we use memory and association of ideas, how we fill in gaps in knowledge and improvise from a few given facts. But exactly how we do it in terms of nerve cell actions we don't know. We are particularly ignorant of the very point that you mentioned just now, Newman, the actual physiology of the pounce on an idea, of the sudden inspiration. Thinking is clearly a motor activity of the brain's cells, a suggestion supported by the common experience that so many people think better with a pen in their hand than viva voce or by reverie and reflection. But you can't so far produce ideas in a man's mind by stimulating his exposed brain here or there electrically. It would have been really exciting if one could have done that - if one could have perhaps excited original thoughts by local stimulation.

It can't be done. Nor does the electro-encephalograph show us how the process of thinking is carried out. It can't tell you what a man is thinking about. We can trace the course, say, of a page of print or of a stream of words into the brain, but we eventually lose them. If we could follow them to their storage places we still couldn't see how they are reassembled later as ideas. You have the great advantage of knowing how your machine was made. We only know that we have in the human nervous system a concern compact in size and in its way perfect for its job. We know a great deal about its microscopical structure and its connections. In fact, we know everything except how these myriads of cells allow us to think. But, Newman, before we say "not only does this machine think but also here in this machine we have an exact counterpart of the wiring and circuits of human nervous systems," I ought to ask whether machines have been built or could be built which are as it were anatomically different, and yet produce the same work.

<u>NEWMAN</u>: The logical plan of all of them is rather similar, but certainly their anatomy, and I suppose

you could say their physiology, varies a lot?

JEFFERSON: Yes, that's what I imagined - we cannot then assume that any one of these electronic machines is a replica of part of a man's brain even though the result of its actions has to be conceded as thought. The real value of the machine to you is its end results, its performance, rather than that its plan reveals to us a model of our brains and nerves. Its usefulness lies in the fact that electricity travels along wires 2 or 3 million times faster than nerve impulses pass along nerves. You can set it to do things that man would need thousands of lives to complete. But that old slow coach, man, is the one with the ideas - or so I think. It would be fun some day, Turing, to listen to a discussion, say on the Fourth Programme, between two machines on why human beings think that they think!

Please return to R. O. Gandy
University College
Leicester

NOT CHECKED IN TALKS DEPARTMENT WITH 'AS BROADCAST' SCRIPT

Producer: T. S. Gregory

AUTOMATIC
CAN/CALCULATING MACHINES BE SAID TO THINK?

Discussion between:-

M. H. A. Newman, F.R.S.,
Professor of Mathematics in the University of Manchester.

A. M. Turing, F.R.S.,
Lecturer in Mathematics in the University of Manchester.

Sir Geoffrey Jefferson, F.R.S.,
Professor of Neuro-Surgery in the University of Manchester.

R. B. Braithwaite, Fellow of King's College, Cambridge.

Recording: Thursday, January 10, 1952.
Disc No.: SLO.1578 Duration:

Transmission: Monday, January 14, 1952. 9.35 - 10.20 THIRD PROGRAMME

Repeat: Wednesday, January 23, 1952. 6.20 - 7.5 THIRD PROGRAMME

BRAITHWAITE: We're here today to discuss whether calculating machines can be said to think in any proper sense of the word. Thinking is ordinarily regarded as so much a speciality of man, and perhaps of other higher animals, that the question may seem too absurd to be discussed. But, of course, it all depends on what is to be included in thinking. The word is used to cover a multitude of different activities. What would you, Jefferson, as a physiologist, say were the most important elements involved in thinking?

JEFFERSON: I don't think that we need waste much time on definition of thinking since it will be hard to get beyond phrases in common usage, such as having ideas in the mind, cogitating, meditating, deliberating, solving problems or imagining. Philologists say that the word "Man" is derived from a Sanskrit word that means "to think," probably in the sense of judging between one idea and another. I agree that we could no longer use the word thinking in a sense that restricted it to man. No one would deny that many animals think, though in a very limited way. They lack insight. For example, a dog learns that it is wrong to get on cushions or chairs with muddy paws, but he only learns it as a venture that doesn't pay. He has no conception of the real reason, that he damages fabrics by doing that.

The average person would perhaps be content to define thinking in very general terms such as revolving ideas in the mind, of having notions in one's head, of having one's mind occupied by a problem and so on. But it is only right to add that our minds are occupied much of the time with trivialities One might say in the end that thinking was the general result of having a sufficiently complex nervous system. Very simple ones do not provide the creature with any problems that are not answered by simple reflex mechanisms. Thinking then becomes all the things that go on in one's brain, things that often end in an action but don't necessarily do so. I should say that it was the sum total of what the brain of man or animal does.

Turing, what do you think about it? Have you a mechanical definition?

<u>TURING</u>: I don't want to give a definition of thinking, but if I had to I should probably be unable to say anything more

about it than that it was a sort of buzzing that went on inside my head. But I don't really see that we need to agree on a definition at all. The important thing is to try to draw a line between the properties of a brain, or of a man, that we want to discuss, and those that we don't. To take an extreme case, we are not interested in the fact that the brain has the consistency of cold porridge. We don't want to say "This machine's quite hard, so it isn't a brain, and so it can't think." I would like to suggest a particular kind of test that one might apply to a machine. You might call it a test to see whether the machine thinks, but it would be better to avoid begging the question, and say that the machines that pass are (let's say) Grade A. machines. The idea of the test is that the machine has to try and pretend to be a man, by answering questions put to it, and it will only pass if the pretence is reasonably convincing. A considerable proportion of a jury, who should not be expert about machines, must be taken in by the pretence. They aren't allowed to see the machine itself - that would make it too easy. So the machine is kept in a far away room and the jury are allowed to ask it questions, which are transmitted through to it: it sends back a typewritten answer.

BRAITHWAITE: Would the questions have to be sums, or could I ask it what it had had for breakfast?

TURING: Oh yes, anything. And the questions don't really have to be questions, any more than questions in a law court are really questions. You know the sort of thing. "I put it to you that you are only pretending to be a man" would be quite in order. Likewise the machine would be permitted all sorts of tricks so as to appear more man-like, such as waiting a bit before giving the answer, or making spelling mistakes, but it can't make smudges on the paper, any more than one can send smudges by telegraph. We had better suppose that each jury has to judge quite a number of times, and that sometimes they really are dealing with a man and not a machine. That will prevent them saying "It must be a machine" every time without proper consideration.

Well, that's my test. Of course I am not saying at present either that machines really could pass the test, or that they couldn't. My suggestion is just that this is the question we should discuss. It's not the same as "Do machines think," but it seems near enough for our present purpose, and raises much the same difficulties.

<u>NEWMAN</u>: I should like to be there when your match between a man and a machine takes place, and perhaps to try my hand at making up some of the questions. But that will be a long time from now, if the machine is to stand any chance with no questions barred?

<u>TURING</u>: Oh yes, at least 100 years, I should say.

<u>JEFFERSON</u>: Newman, how well would existing machines stand up to this test? What kind of things can they do now?

<u>NEWMAN</u>: Of course their strongest line is mathematical computing, which they were designed to do, but they would also do well at some questions that don't look numerical, but can easily

be made so, like solving a chess problem or looking you up a train in the time-table.

BRAITHWAITE: Could they do that?

NEWMAN: Yes. Both these jobs can be done by trying all the possibilities, one after another. The whole of the information in an ordinary time-table would have to be written in as part of the programme, and the simplest possible routine would be one that found the trains from London to Manchester by testing every train in the time-table to see if it calls at both places, and printing out those that do. Of course this is a dull plodding method, and you could improve on it by using a more complicated routine, but if I have understood Turing's test properly, you are not allowed to go behind the scenes and criticise the method, but must abide by the scoring on correct answers, found reasonably quickly.

JEFFERSON: Yes, but all the same a man who has to look up trains frequently gets better at it, as he learns his way about the time-table. Suppose I give a machine the same problem again, can it learn to do better without going through the whole rigmarole of trying everything over every time? I'd like to have your answer to that because it's such an important point. Can machines learn to do better with practice?

NEWMAN: Yes, it could. Perhaps the chess problem provides a better illustration of this. First I should mention that all the information required in any job - the numbers, times of trains, positions of pieces, or whatever it is, and also the instructions saying what is to be done with them - all this material is stored in the same way. (In the Manchester machine it is stored as a pattern on something resembling a television screen.) As the work goes on the pattern is changed. Usually it is the part of the pattern that contains the data that changes, while the instructions stay fixed. But it is just as simple to arrange that the instructions themselves shall be changed now and then. Well, now a programme could be composed that would cause the machine to do this: a 2-move chess problem is recorded into the machine in some suitable coding, and whenever the machine is started, a white move is chosen at random (there is a device for making random choices in our machine). All the consequences of this move are now analysed, and if it does not lead to forced mate in two moves, the machine prints, say, "P-Q3, wrong move," and stops. But the analysis shows that when the right move is chosen the machine not only prints, say, "B-Q5, solution," but it changes the instruction calling for a random choice to one that says "Try B-Q5." The result is that whenever the machine is started again it will immediately print out the right solution - and this without the man who made up the routine knowing beforehand what it was. Such a routine could certainly be made now, and I think this can fairly be called learning.

JEFFERSON: Yes, I suppose it is. Human beings learn by repeating the same exercises until they have perfected them. Of course it goes further and at the same time - we learn generally to shift the knowledge gained about one thing to another set of problems, seeing relevances and relationships. Learning means remembering. How long can a machine store information for?

NEWMAN: Oh, at least as long as a man's lifetime, if it is refreshed occasionally.

JEFFERSON: Another difference would be that in the learning process there is much more frequent intervention by teachers, parental or otherwise, guiding the arts of learning. You mathematicians put the programme once into the machine and leave it to it. You wouldn't get any distance at all with human beings if that is what you did. In fact, the only time you do that in the learning period is at examinations.

TURING: It's quite true that when a child is being taught, his parents and teachers are repeatedly intervening to stop him doing this or encourage him to do that. But this will not be any the less so when one is trying to teach a machine. I have made some experiments in teaching a machine to do some simple operation, and a very great deal of such intervention was needed before I could get any results at all. In other words the machine learnt so slowly that it needed a great deal of teaching.

JEFFERSON: But who was learning, you or the machine?

TURING: Well, I suppose we both were. One will have to find out how to make machines that will learn more quickly if there is to be any real success. One hopes too that there will be a sort of snowball effect. The more things the machine has learnt the easier it ought to be for it to learn others. In learning to do any particular thing it will probably also be learning to learn more efficiently. I am inclined to believe that when one has taught it to do certain things one will find that some other things which one had planned to teach it are happening without any special teaching being required. This certainly happens with an intelligent human mind, and if it doesn't happen when one is teaching a machine there is something lacking in the machine. What do you think about learning possibilities, Braithwaite?

<u>BRAITHWAITE</u>: No one has mentioned what seems to me the great difficulty about learning, since we've only discussed learning to solve a particularproblem. But the most important part of human learning is learning from experience - not learning from one particular kind of experience, but being able to learn from experience in general. A machine can easily be constructed with a feed-back device so that the programming of the machine is controlled by the relation of its output to some feature in its external environment - so that the working of the machine in relation to the environment is self-corrective. But this requires that it should be some particular feature of the environment to which the machine has to adjust itself. The peculiarity of men and animals is that they have the power of adjusting themselves to almost all the features. The feature to which adjustment is made on a particular occasion is the one the man is attending to and he attends to what he is <u>interested in.</u>

His interests are determined, by and large, by his appetites, desires, drives, instincts - all the things that together make up his "springs of action." If we want to construct a machine which will vary its attention to things in its environment so that it will sometimes adjust itself to one and sometimes to another, it would seem to be necessary to equip the machine with something corresponding to a set of appetites. If the machine is built to be treated only as a domestic pet, and is spoon-fed with particular problems, it will not be able to learn in the varying way in which human beings learn. This arises from the necessity of adapting behaviour suitably to environment if human appetites are to be satisfied.

JEFFERSON: Turing, you spoke with great confidence about what you are going to be able to do. You make it sound as if it would be fairly easy to modify construction so that the machine reacted more like a man. But I recollect that from the time of Descartes and Borelli on people have said that it would be only a matter of a few years, perhaps 3 or 4 or maybe 50, and a replica of man would have been artifically created. We shall be wrong, I am sure, if we give the impression that these things would be easy to do.

NEWMAN: I agree that we are getting rather far away from computing machines as they exist at present. These machines have rather restricted appetites, and they can't blush when they're embarrassed, but it's quite hard enough, and I think a very interesting problem, to discover how near these actually existing machines can get to thinking. Even if we stick to the reasoning side of thinking, it is a long way from solving chess problems to the invention of new mathematical concepts or making a generalisation that takes in ideas that were current before, but had never been brought together as instances of a single general notion.

BRAITHWAITE: For example?

NEWMAN: The different kinds of number. There are the integers, 0, 1, -2, and so on; there are the real numbers used in comparing lengths, for example the circumference of a circle and its diameter; and the complex numbers involving $\sqrt{-1}$; and so on. It is not at all obvious that these are instances of one thing, "number." The Greek mathematicians used entirely different words for the integers and the real numbers, and had no single idea to cover both. It is really only recently that the general notion of kinds of number, had been abstracted from these instances and accurately defined. To make this sort of generalisation you need to have the power of recognising similarities, seeing analogies between things that had not been put together before. It is not just a matter of testing things for a specified property and classifying them accordingly. The concept itself has to be framed, something has to be created, say the idea of a number-field. Can we even guess at the way a machine could make such an invention from a programme composed by a man who had not the concept in his own mind?

TURING: It seems to me, Newman, that what you said about 'trying out possibilities' as a method applies to quite an extent, even when a machine is required to do something as advanced as finding a useful new concept. I wouldn't like to have to define the meaning of the word "concept," nor to give rules for rating their usefulness, but whatever they are they've got outward and visible forms, which are words and combinations of words. A machine could make up such combinations of words more or less at random, and then give them marks for various merits.

NEWMAN: Wouldn't that take a prohibitively long time?

TURING: It would certainly be shockingly slow, but it could start on easy things, such as lumping together rain, hail, snow and sleet, under the word 'precipitation.' Perhaps it might do more difficult things later on if it was learning all the time how to improve its methods.

BRAITHWAITE: I don't think there's much difficulty about seeing analogies that can be formally analysed and explicitly stated. It is then only a question of designing the machine so that it can recognise similarities of mathematical structure. The difficulty arises if the analogy is a vague one about which little more can be said than that one has a feeling that there is some sort of similarity between two cases but one hasn't any idea as to the respect in which the two cases are similar. A machine can't recognise similarities when there is nothing in its programme to say what are the similarities it is expected to recognise.

TURING: I think you could make a machine spot an analogy, in fact it's quite a good instance of how a machine could be made to do some of those things that one usually regards as essentially a human monopoly. Suppose that someone was trying to explain the double negative to me, for instance, that when something isn't not green it must be green, and he couldn't quite get it across. He might say 'Well, it's like crossing the road. You cross it, and then you cross it again, and you're back where you started.' This remark might just clinch it. This is one of the things one would like to work with machines, and I think it would be likely to happen with them. I imagine that the way analogy works in our brains is something like this. When two or more sets of ideas have the same pattern of logical connections, the brain may very likely economise parts by using some of them twice over, to remember the logical connections both in the one case and in the other. One must suppose that some part of my brain was used twice over in this way, once for the idea of double negation and once for crossing the road, there and back. I am really supposed to know about both these things but can't get what it is the man is driving at, so long as he is talking about all those dreary nots and not-nots. Somehow it doesn't get through to the right part of the brain. But as soon as he says his piece about crossing the road it gets through to the right part, but by a different route. If there is some such purely mechanical explanation of how this argument by analogy goes on in the brain, one could make a digital computer do the same.

<u>JEFFERSON</u>: Well, there isn't a mechanical explanation in terms of cells and connecting fibres in the brain.

BRAITHWAITE: But could a machine really do this? How would it do it?

TURING: I've certainly left a great deal to the imagination. If I had given a longer explanation I might have made it seem more certain that what I was describing was feasible, but you would probably feel rather uneasy about it all, and you'd probably exclaim impatiently, 'Well, yes, I see that a machine could do all that, but I wouldn't call it thinking.' As soon as one can see the cause and effect working themselves out in the brain, one regards it as not being thinking, but a sort of unimaginative donkey-work. From this point of view one might be tempted to define thinking as consisting of "those mental processes that we don't understand." If this is right then to make a thinking machine is to make one which does interesting things without our really understanding quite how it is done.

JEFFERSON: If you mean that we don't know the wiring in men, as it were, that is quite true.

TURING: No, that isn't at all what I mean. We know the wiring of our machine, but it already happens there in a limited sort of way. Sometimes a computing machine does do something rather weird that we hadn't expected. In principle one could have predicted it, but in practice it's usually too much trouble. Obviously if one were to predict everything a computer was going to do one might just as well do without it.

NEWMAN: It is quite true that people are disappointed when they discover what the big computing machines actually do, which is just to add and multiply, and use the results to decide what further additions and multiplications to do. 'That's not thinking', is the natural comment, but this is rather begging the question. If you go into one of the ancient churches in Ravenna you see some most beautiful pictures round the walls, but if you peer at them through binoculars you might say, 'Why, they aren't really pictures at all, but just a lot of little coloured stones with cement in between.' The machines processes are mosaics of very simple standard parts, but the designs can be of great complexity, and it is not obvious where the limit is to the patterns of thought they could imitate.

BRAITHWAITE: But how many stones are there in your mosaic? Jefferson, is there a sufficient multiplicity of the cells in the brain for them to behave like a computing machine?

JEFFERSON: Yes, there are thousands, tens of thousands more cells in the brain than there are in a computing machine, because the present machine contains - how many did you say?

TURING: Half a million digits. I think we can assume that is the equivalent of half a million nerve cells.

BRAITHWAITE: If the brain works like a computing machine then the present computing machine cannot do all the things the brain does. Agreed; but if a computing machine were made that could do all the things the brain does, wouldn't it require more digits than there is room for in the brain?

<u>JEFFERSON</u>: Well, I don't know. Suppose that it is right to equate digits in a machine with nerve cells in a brain. There are various estimates, somewhere between ten thousand million and fifteen thousand million cells are supposed to be there. Nobody knows for certain, you see. It is a colossal number. You would need 20,000 or more of your machines to equate digits with nerve cells. But it is not, surely, just a question of size. There would be too much logic in your huge machine. It wouldn't be really like a human output of thought. To make it more like, a lot of the machine parts would have to be designed quite differently to give greater flexibility and more diverse possibilities of use. It's a very tall order indeed.

<u>TURING</u>: It really is the size that matters in this case. It is the amount of information that can be stored up. If you think of something very complicated that you want one of these machines to do, you may find the particular machine you have got won't do, but if any machine can do it at all, then it can be done by your first computer, simply increased in its storage capacity.

<u>JEFFERSON</u>: If we are really to get near to anything that can be truly called 'thinking' the effects of external stimuli cannot be missed out; the intervention of all sorts of extraneous factors, like the worries of having to make one's living, or pay one's taxes, or get food that one likes. These are not in any sense minor factors, they are very important indeed, and worries concerned with them may greatly interfere with good thinking, especially with

creative thinking. You see a machine has no environment, and man is in constant relation to his environment, which, as it were punches him whilst he punches back. There is a vast background of memories in a man's brain that each new idea or experience has to fit in with. I wonder if you could tell me how far a calculating machine meets that situation. Most people agree that man's first reaction to a new idea (such as the one we are discussing today) is one of rejection, often immediate and horrified denial of it. I don't see how a machine could as it were say "now Professor Newman or Mr. Turing, I don't like this programme at all that you've just put it into me, in fact I'm not going to have anything to do with it!"

NEWMAN: One difficulty about answering that is one that Turing has already mentioned. If someone says, "Could a machine do this, e.g. could it say 'I don't like the programme you have just put into me'," and a programme for doing that very thing is duly produced, it is apt to have an artificial and ad hoc air, and appear to be more of a trick than a serious answer to the question. It is like those passages in the Bible, which worried me as a small boy, that say that such and such was done "that the prophecy might be fulfilled which says" so and so. This always seemed to me a most unfair way of making sure that the prophecy came true. If I answer your question, Jefferson, by making a routine which simply caused the machine to say just the words "Newman and Turing, I don't like your programme," you would certainly feel this was a rather childish trick, and not the answer to what you really wanted to know. But yet it's hard to pin down what you do want.

JEFFERSON: I want the machine to reject the problem because it offends it in some way. That leads me to enquire what the ingredients are of ideas that we reject because we instinctively don't care for them. I don't know why I like some pictures and some music and am bored by other sorts. But I'm not going to carry that line on because we are all different, our dislikes are based on our personal histories and probably too on small differences of construction in all of us, I mean by heredity. Your machines have no genes, no pedigrees. Mendelian inheritance means nothing to wireless valves. But I don't want to score debating points! We ought to make it clear that not even Turing thinks that all that he has to do is to put a skin on the machine and that it is alive! We've been trying for a more limited objective whether the sort of thing that machines do can be considered as thinking. But is not your machine more certain than any human being of getting its problem right at once, and infallibly?

NEWMAN: Oh !

TURING: Computing machines aren't really infallible at all. Making up checks on their accuracy is quite an important part of the art of using them. Besides making mistakes they sometimes haven't done quite the calculation one had expected, and one gets something that might be called a "misunderstanding."

<u>JEFFERSON</u>: At any rate, they are not influenced by the emotions. You have only to upset a person enough and he becomes confused, he can't think of the answers and may make a fool of himself. It is high emotional content of mental processes in the human being that makes him quite different from a machine. It seems to me to come from the great complexity of his nervous system with its 10^{10} cells and also from his endocrine system which imports all sorts of emotions and instincts, such as those to do with sex. Man is essentially a chemical machine, he is much affected by hunger and fatigue, by being "out of sorts" as we say, also by innate judgments, and by sexual urges. This chemical side is tremendously important, not the least so because the brain does exercise a remote control over the most important chemical processes that go on in our bodies. Your machines don't have to bother with that, with being tired or cold or happy or satisfied. They show no delight at having done something never done before. No, they are "mentally" simple things. I mean that however complicated their structure is (and I know it <u>is</u> very complicated) compared with man they are very simple and perform their tasks with an absence of distracting thoughts which is quite <u>inhuman</u>.

<u>BRAITHWAITE</u>: I'm not sure that I agree. I believe that it will be necessary to provide the machine with something corresponding to appetites, or other "springs of action," in order that it will pay enough attention to relevant features in its environment to be able to learn from experience. Many psychologists have held that the

emotions in men are by-products of their appetites and that they serve a biological function in calling higher levels of mental activity into play when the lower levels are incapable of coping with an external situation. For example, one does not feel afraid when there is no danger, or a danger which can be avoided more or less automatically: fear is a symptom showing that the danger has to be met by conscious thought. Perhaps it will be impossible to build a machine capable of learning in general from experience without incorporating in it an emotional apparatus, the function of which will be to switch over to a different part of the machine when the external environment differs too much from what would satisfy the machines appetites by more than a certain amount. I don't want to suggest that it will be necessary for the machine to be able to throw a fit of tantrums. But in humans tantrums frequently fulfil a definite function - that of escaping from responsibility; and to protect a machine against a too hostile environment it may be essential to allow it, as it were, to go to bed with a neurosis, or psychogenic illness - just as, in a simpler way, it is provided with a fuse to blow, if the electric power working it threatens its continued existence.

TURING: Well, I don't envisage teaching the machine to throw temperamental scenes. I think some such effects are likely to occur as a sort of by-product of genuine teaching, and that one will be more interested in curbing such displays than in encouraging them. Such effects would probably be distinctly different from the corresponding human ones, but recognisable as variations on them. This means that if the machine was being put through one of my imitation tests, it would have to do quite a bit of acting, but if one was comparing it with a man in a less strict sort of way the resemblance might be quite impressive.

NEWMAN: I still feel that too much of our argument is about what hypothetical future machines will do. It is all very well to say that a machine could easily be made to do this or that, but, to take only one practical point, what about the time it would take to do it? It would only take an hour or two to make up a routine to make our Manchester machine analyse all possible variations of the game of chess right out, and find the best move that way - if you didn't mind its taking thousands of millions of years to run through the routine. Solving a problem on the machine doesn't mean finding a way to do it between now and eternity, but within a reasonable time. This is not just a technical detail that will be taken care of by future improvements. It's most unlikely that the engineers can ever give us a factor of more than a thousand or two times our present speeds. To assume that runs that would take thousands of millions of years on our present machines will be done in a flash on machines of the future, is to move into the realms of science fiction.

TURING: To my mind this time factor is the one question which will involve all the real technical programming difficulty. If one didn't know already that these things can be done by brains within a reasonable time one might think it hopeless to try with a machine. The fact that a brain can do it seems to suggest that the difficulties may not really be so bad as they now seem.

BRAITHWAITE: I agree that we ought not to extend our discussion to cover whether calculating machines could be made which would do everything that a man can do. The point is, surely, whether they can do all that it is proper to call thinking. Appreciation of a picture contains elements of thinking, but it also contains elements of feeling; and we're not concerned with whether a machine can be made that will feel. Similarly with moral questions: we're only concerned with them so far as they are also intellectual ones. We haven't got to give the machine a sense of duty or anything corresponding to a will: still less need it be given temptations which it would then have to have an apparatus for resisting. All that it has got to do in order to think is to be able to solve, or to make a good attempt at solving, all the intellectual problems with which it might be confronted by the environment in which it finds itself. This environment, of course, must include Turing asking it awkward questions as well as natural events such as being rained upon, or being shaken up by an earthquake.

NEWMAN: But I thought it was you who said that a machine wouldn't be able to learn to adjust to its environment if it hadn't been provided with a set of appetites and all that went with them?

BRAITHWAITE: Yes, certainly. But the problems raised by a machine having appetites are not properly our concern today. It may be the case that it wouldn't be able to learn from experience without them; but we're only required to consider whether it would be able to learn at all - since I agree that being able to learn is an essential part of thinking. So oughtn't we to get back to something centred on thinking? Can a machine make up new concepts, for example?

NEWMAN: There are really two questions that can be asked about machines and thinking, first, what do we require before we agree that the machine does everything that we call thinking? This is really what we have been talking about for most of the time; but there is also another interesting and important question: Where does the doubtful territory begin? What is the nearest thing to straight computing that the present machines perhaps can't do?

BRAITHWAITE: And what would your own answer be?

NEWMAN: I think perhaps to solve mathematical problems for which no method is known, in the way that men do; to find new methods. This is a much more modest aim than inventing new mathematical concepts. What happens when you try to solve a new problem in the ordinary way is that you think about it for a few seconds, or a few years, trying out all the analogies you can think of with problems that have been solved, and then you have an idea. You try it out in detail. If it is no good you must wait for another idea. This is a little like the chess-problem routine, where one move after another is tried, but with one very important difference, that if I am even a moderately good mathematician the ideas that I get are not just random ones, but are pre-selected so that there is an appreciable chance that after a few trials one of them will be successful. Henry Moore says about the studies he does for his sculpture, "When the work is more than an exercise, inexplicable jumps occur. This is where the imagination comes in." If a machine could really be got to imitate this sudden pounce on an idea, I believe that everyone would agree that it had begun to think, even though it didn't have appetites or worry about the income tax. And suppose that we also stuck to what we know about the physiology of human thinking, how much would that amount to, Jefferson?

JEFFERSON: We know a great deal about the end-product, thinking itself. Are not the contents of our libraries and museums the total up to date? Experimental psychology has taught us a lot about the way that we use memory and association of ideas, how we fill in gaps in knowledge and improvise from a few given facts. But exactly how we do it in terms of nerve cell actions we don't know. We are particularly ignorant of the very point that you mentioned just now, Newman, the actual physiology of the pounce on an idea, of the sudden inspiration. Thinking is clearly a motor activity of the brain's cells, a suggestion supported by the common experience that so many people think better with a pen in their hand than viva voce or by reverie and reflection. But you can't so far produce ideas in a man's mind by stimulating his exposed brain here or there electrically. It would have been really exciting if one could have done that - if one could have perhaps excited original thoughts by local stimulation. It can't be done. Nor does the electro-encephalograph show us how the process of thinking is carried out. It can't tell you what a man is thinking about. We can trace the course, say, of a page of print or of a stream of words into the brain, but we eventually lose them. If we could follow them to their storage places we still couldn't see how they are reassembled later as ideas. You have the great advantage of knowing how your machine was made. We only know that we have in the human nervous system a concern compact in size and in its way perfect for its job. We know a great deal about its microscopical structure and its connections. In fact, we know everything except how those myriads of cells allow us to think. But, Newman, before we say "not only does this machine think but also here in this machine we have an exact counterpart of the wiring and circuits of human nervous systems," I ought to ask whether machines have been built or could be built which are as it were anatomically different, and yet produce the same work.

<u>NEWMAN</u>: The logical plan of all of them is rather similar, but certainly their anatomy, and I suppose you could say their physiology, varies a lot?

<u>JEFFERSON</u>: Yes, that's what I imagined - we cannot then assume that any one of these electronic machines is a replica of part of a man's brain even though the result of its actions

has to be conceded as thought. The real value of the machine to you is its end results, its performance, rather than that its plan reveals to us a model of our brains and nerves. Its usefulness lies in the fact that electricity travels along wires 2 or 3 million times faster than nerve impulses pass along nerves. You can set it to do things that man would need thousands of lives to complete. But that old slow coach, man, is the one with the ideas - or so I think. It would be fun some day, Turing, to listen to a discussion, say on the Fourth Programme, between two machines on why human beings think that they think!

mjl