AG AboutGender
International journal of gender studies

http://www.aboutgender.unige.it

Vol. 5 N° 9 anno 2016
pp. 103-122

Binary codes. A gender-informed discussion on professionalism in nascent digital computing

Mariacristina Sciannamblo

Università di Roma Sapienza

Abstract

This article explores the connections between two analytical concerns within the field of history and social studies of science and technology, namely the demand to reform the history of computing on the one hand, and the use of 'gender' as analytical category on the other. I shall bring into focus the "question of professionalism" (Ensmenger 2001) in computer fields as a crucial point through which to shed light on the controversial role of women in computing industry, along with the benefit of aligning the history of computing with feminist perspectives.

The article focuses on the dawn of digital computing era in the USA by discussing the work of the first women programmers behind the Electronic Numerical Integrator And Computer (ENIAC). The argument that I advance is that the attempts to build computer work as a professional field and expertise are in many respects biased by gendered discursive and material practices.

Keywords: digital computing, gender, professionalism, women, expertise.

1. Introduction

In a recent column titled *The Tears of Donald Knuth*, Thomas Haigh (2015) addresses under renovate spirit some controversial issues regarding the relationship between the discipline of computer science and the growing body of work on the history of computing[1]. The article takes shape from a talk given by the famous computer scientist Donald Knuth at Stanford University[2]. In his reflections, Haigh challenges Knuth's complaint that historians are following a discouraging trend in doing history of computer science by ignoring technical details and, in so doing, "dumbing down" their mission and outcomes. According to Haigh, the reasons why little technical history of computer science has been produced by trained historians rely upon disciplinary and institutional factors, common both to History of Science and Computer Science fields. Nevertheless, Haigh adds, the blossom of a notable amount of historical works that draw explicit connections between the history of computing and computer science pursues a holistic approach that aims at integrating technical analysis and the attention to social, institutional, cultural and political factors[3]. Furthermore, in recalling that «computing is much bigger than computer science, and so the history of computing is much bigger than the history of computer science», Haigh emphasizes the fact that historical analyses focused on a broad range of computer-related fields, such as business, cybernetics, semiconductor industry, punched card machines, IT workforce, personal computer, the use of computer in developed countries and in some medical practices, are equally respectable examples of work, other than computer science.

Bearing in mind these concerns, I discuss here the issue of 'professionalism' in computer work, highlighting some interesting remarks regarding both the field of

[1] The article is available at: http://cacm.acm.org/magazines/2015/1/181633-the-tears-of-donald-knuth/fulltext
[2] Available at: https://www.youtube.com/watch?v=gAXdDEQveKw
[3] All the references are mentioned in Haigh's article.

computer science and that of computing labor. I shall argue that the attempts to define computer work as a profession and technical expertise unveil remarkable gender biases that affected both the composition of the workforce and the birth of computing as organizational culture. Following a chronological order, in the first paragraph I shall reach back to the birth of the electronic computing era by shedding light on the important contributions of the first female programmers at work on the ENIAC project. I shall bring into focus the contradiction between the perception of female labor by both male managers and media reports, and the effective, compound operations they carried out. In the second paragraph, I shall outline some emergent patterns of change in the conception of computer programming during the 1950s and the 1960s, with particular emphasis on the idea of programming as a “black art”, combining both analytic skills and creativity. The growth of the industry, thus the increase in economic and social interests around new computer jobs, fostered the emergence of two phenomena that shape each other: the devaluation of female labor on the one hand, the gendered debate on professionalism between the nascent computer science and the pragmatic approach to computer work on the other, which are illustrated in the third paragraph.

Lastly, I shall recall the importance of aligning historical analysis and sociological investigation with feminist critique of science and technology. I argue, indeed, that such an analytical move is crucial in order to understand that pressing social issues such as gender inequalities in IT educational paths and careers require researchers to go beyond the assessment of figures and numbers so as to call into question the alleged “neutral” nature of technical expertise, professional status and labor organization.

2. Where are the women? The feminization of labor in the nascent digital computing

The shortage of female workforce in IT industry and education is increasingly becoming a sensible issue, concerning both academic scholarship (see Margolis and Fisher 2002; Misa 2010; Abbate, 2012) and policy makers[4]. In order to investigate the

[4] For a critical analysis of the term ‘Information Technology’, see Kline 2006.

profound roots of this phenomenon, it is important to go back to the early digital electronic computing era and to look at the role of female labor in the rising industry.

There are several historical studies that have pointed out the prominent work of women in computer industry, both in the USA and UK. In an essay eloquently titled *When Computers Were Women*, Jennifer S. Light (1999) engages in retelling the development of the first general-purposes electronic computer, the Electronic Numerical Integrator And Computer (ENIAC), by shedding light on the large amount of women that worked as proto-programmers, very close to those engineers regarded as pioneers in the history of computing such as J. Presper Eckert and John W. Mauchly. The main thesis advanced by Light is that the historiography of computing and computer labor has repeatedly dismissed the presence and the value of female work, fostering, that way, the popular image and belief of the programmer as a male job. However, a closer look at the dynamics that animated the nascent computing industry during wartime brings up more nuanced questions regarding the women's employment: what is the nature of work undertaken by female employees at the time? What was the perception of female labor at the beginning of computing industry? What were the effective conditions behind the high number of women employed in computing careers? In addressing these queries, we can realize the pervasiveness of gendered assumptions and practices in computing industry, capable of shaping the nature of its expertise, organization of work and the purposes of computing itself.

The outbreak of World War II engendered important changes in the US job market, since male workers were drafted into the army. Women were encouraged to apply for technical jobs, mostly concerning the assistant level. As Light underlines, in fact, apart from the women with a Ph.D. degree, the rest of the female workforce was intended as temporary, without any chance to climb up the job ladder. In *Recoding Gender Women's Changing Participation in Computing*, Janet Abbate (2012) provides a reliable account about the early women programmers, remarking that the fundamental reason why women came to staff the first electronic digital computer was the lack of male manpower due to the war. It was just a contingency indeed, with the understanding that women would vacate those positions after the return of men, so as to restore the traditional gendered division of labor. Labor patterns in scientific and clerical occupations are, in fact, at the base of the paradox that several scholars have recalled as characteristic of the role of women in early computer jobs (Light 1999; Abbate 2012, 2012; Payette 2014). The paradox consists in the complexity and degree of innovation conveyed by women's work and the tendency from higher hierarchies and media to depict the same jobs as "unprofessional". Although the members of the female workforce hired within the ENIAC project were not even recognized as individual identities, but commonly known as "ENIAC girls", they took on tasks that demanded high levels of mathematical skills and, at the same time, they were downgraded as "subprofessional" (Light 1999). Although the ENIAC was designed to take over scientific calculations carried out by humans until then (Grier 2005), it was necessary to provide a certain amount of human labor as far as programming equations into the machine was concerned, a task undertaken by those human beings now called 'operators'. They mainly work on ballistic studies with desk calculators and differential analyzers to program, handling complex tasks related to machine's circuitry, logic, physical structure and operation. As Light highlights, the ENIAC project comprised two parts – hardware and software – clearly framed according to gender patterns: working with hardware was considered a men's job, whereas software programming was usually undertaken by women and regarded as a secondary, clerical task.

The mismatch between the actual work performed by women and the terms that employers and media coverage used to categorize it is one of the most controversial issues that is worth investigating, not only to reinstitute fairness into historical analysis, but also to provide a more nuanced understanding of sociotechnical processes, such as the gendered division of labor and the creation of specific expertise whereby the actors have constructed the technological frames (Bijker 1987) related to the nascent electronic computer. In an article published in 1996 in the Annals of the History of Computing, W. Barkley Fritz reports the stories of the women behind the development of the ENIAC between 1942 and 1955, spanning from the wartime to its full usage, through the period of its design and conversion. Reading the direct accounts of the female protagonists who participated in the development and launch of the "machine that changed the world" (Fritz 1996, 13), as media reports depicted the birth of the digital computing era, is interesting when acknowledging the historical circumstances that led to the design of the ENIAC, roles and hierarchies in the workplace, career paths for men and women as well as the gendered material practices and knowledge behind the hardware and software of the computational machine.

As several scholars recall (Fritz 1996; Light 1999; Abbate 2012), the original team at work on the new project comprised six female coders: Kathleen McNulty, Frances Bilas, Betty Jean Jennings, Elizabeth Snyder, Ruth Lichterman, and Marlyn Wescoff[5]. Their stories had slightly different traits as to their backgrounds, but also some common points such as, for example, the refusal to teach mathematics in secondary school or to do repetitive calculations for insurance companies as well as their excitement for programming and being part of a novel adventure[6]. They had all been hired by the Moore School of Electrical Engineering of the University of Pennsylvania, where they worked with the differential analyzer and desk calculators to compute trajectories for artillery firing tables. As they referred (Fritz 1996), initially none of them knew about the new project despite the fact that they were required to undertake complex tasks, improve their theoretical knowledge and work many hours per day. Moreover, their occupation as computer programmers was considered "SP-4", a subprofessional civil service grade. It is worth quoting the words of Kathleen McNulty, describing the work with the ENIAC:

> Operation included setting up the boundary conditions in the integrators, repairing or replacing the strings and bands on the torque amplifiers, guiding the arbitrary functions from input tables, and punching out the results of the calculations at specified times and at summit and ground. These two men and a young woman trained Fran [Frances Bilas] and me as operators for the differential analyzer, so that in a short time we were able to take over a work shift. We worked from 8 a.m. until 4:30 p.m. for two weeks, then changed over to 4 p.m. to 11:30 p.m. for two weeks. (Fritz 1996, 16)

---

[5] As Nathan Ensmenger (2010) notes, the ENIAC girls are widely considered the first computer programmers, but, in the 1940s, they worked as coders, meaning that they basically translated into machine language the higher formal mathematical language developed by male scientists and engineers.

[6] As emerged from the direct accounts (Fritz 1996; Abbate 2012), the most common occupations for women with a college degree in mathematics were teaching in high school or work as actuary in insurance companies.

As clarified by these accounts, the nature of work undertaken by female employees required a kind of knowledge and abilities never seen before that time, made up of mathematical analysis, logical reasoning, but also topics outside the maths curriculum such as numerical integration (*ibidem*). The need of an unprecedented and interdisciplinary expertise to set up the new machine went along with the sense of an interesting and unknown adventure for all the operators at work on the ENIAC. Indeed, when the young women moved to the Moore School in Philadelphia, the project was classified, so that only officers were aware of it, requiring trust and commitment more than a specific competence. As Betty Jean Jennings (Bartik) recalled:

> […] an announcement was made that APG was recruiting what would later be known as coder/programmers for ENIAC, a new machine being completed at the Moore School. Anyone who wanted to apply could go to a meeting at the Moore School. I had no idea what the job was or what the ENIAC was. All I knew was that I might be getting in on the ground floor of something new, and I believed I could learn and do anything as well as anyone else. I went to the meeting. There must have been a dozen or so of us. We were told very little about the ENIAC because it was still classified. Each of us was called in for an interview with Herman Goldstine and Leland Cunningham. Dr. Goldstine was the BRL liaison officer with the ENIAC project, and Dr. Cunningham was an astronomer from APG. They asked a few questions, and I remember Herman asking me what I knew about electricity. I said I had had a course in physics and knew E = I/R. He replied what he really wanted to know was, Are you afraid of it? I replied that I wasn't. His wife, Adele, then came into the room and called me by name. (*ibidem*, 19)

After recruitment, the new personnel spent some time at APG, where they learned how to work with the various punch card machines such as tabulator, sorter, reader, reproducer, punch, and how to set up the control boards. This kind of training was prescribed by the situated technology of scientific problem-solving that ruled the function of the machine at the time, before the introduction of high-speed electronic calculators. At the base of the ENIAC's hardware, indeed, there were function tables that contained general mathematical solutions to be computed (by performing tricky arithmetic operations) so as to generate accurate ballistic data helpful to some of the US World War II activities, such as the Manhattan Project.

Although female labor involved tasks of a high degree of innovation, women were considered as mere operators, executors of engineers' instructions and, at the beginning, kept in the dark about the entity of the project. Nevertheless, they quickly became very skillful in programming, learning how the new machine worked through logical diagrams as well as by trials and errors (Light 1999). Thus, they acted as knowledge producers even though their status and role prescribed a subaltern position with reference to male engineers and officers. This gendered division of labor, determined by an "idiom of sex-typing" (Milkman 1987 quoted in Light 1999), suggests anyway that organizational hierarchies and a severe demarcation between knowledge and execution did not reflect the way the computer was designed, especially with regard to its double configuration of hardware and software. The close relationship between the machine's physical structure and the abstract operations that make it work, after all, was already clear with the differential analyzer that characterized the analog-computing era. The story of the development of the analyzer carried out by Vannevar Bush and his colleagues at MIT (Owens 1996) points out how improvements made on hard components (shafts, discs, tables) were associated to as many refinements in mathematical calculations, just as the programming of the ENIAC assigned to women required a complete understanding of the machine's design controlled by men[7].

After the presentation and public demonstration of ENIAC in 1946, many female programmers retired to raise a family, whereas those who chose to hold the job kept on working in subprofessional roles, away from those professional trappings (Ensmenger, 2001) that will emerge in the following years.

[7] The job of programming required skills in trouble-shooting, that, in turn, involved the knowledge of both the application and the machine.

3. From clerical work to poetry: traces of transformation in computer programming

As we have seen in the above section, the early era of digital electronic computing was informed by a strict division of labor, with men undertaking design jobs that involved mostly hardware components, and women carrying out operative task with software applications. No one could ever expect that programming would have been regarded as a "black art" (Ensmenger 2010, 27) over the two decades after the introduction of the electronic computer to the world. This paragraph puts just those 1950s and 1960s under scrutiny, regarding them as years characterized by a fast growing computing industry and a parallel uncertainty regarding professional roles and expertise pertaining to it.

The idea of 'black art' recalls the concept of 'black box' (see Winner 1980; Pinch 1992), namely something that lacks transparency, whose components and functions are difficult to detect clearly. As the stories of ENIAC girls remark, the traditional division of labor, which implied a clear demarcation between clerical and intellectual tasks, did not work with the design of the new machine insofar as it prescribed new paths of organizational management and new job profiles. These issues echo in many respects the sociotechnical dynamics that Wiebe Bijker and Trevor Pinch (1984) describe with regard to the design and development of the bicycle. By illustrating how different paths of technological development have been shaped by the demands of different social groups, diverging interpretations and rhetoric, technical constraints as well as social and historical contingencies, they frame the emergence of technologies as a "process" rather than "an isolated event" (Bijker and Pinch 1984, 416). In addition, it is not simply the different interpretations assigned to the artifact by relevant social groups that matter, but also how different uses are implicated in the design and re-design of the artifact itself. Looking back at the dawn of the commercial electronic digital computing industry with these analytical lenses, we can understand, for instance, why programmers were able – often more than engineers – to scrutinize the vacuum tube technology of the ENIAC as well as its program. Indeed, the interpretation of the machine provided by designers often struggled with the use and knowledge programmers developed along their work. In this respect, there are a few lines from Betty Jean Jennings' account about the tests made before the public demonstration of the ENIAC in February 1946, that are worth being mentioned:

> The night before the demonstration, the trajectory program was running perfectly, *except* it didn't stop computing when it was calculated to hit the ground. It kept going. Betty [Holberton] and I checked and rechecked everything until about 2 a.m. During the night it came to Betty what was wrong. She came in the next morning and flipped one switch on the master programmer and the problem was solved. (Fritz 1996, 21, emphasis in original).

This passage is rather eloquent about the strong tie between software applications and hardware components, which, in turn, defined the functioning of electronic calculators integrated rather than divided into independent technical functions and clerical tasks. Besides, it is worth noting the debt that, at the time, computing owed to electrical engineering before computer science would take over it, sanctioning "an *explicit* division" of hardware and software processes[8] (Ceruzzi 1989, 273, emphasis in original).

Although the new information industry benefited from a great commercial expansion, the computer programmer continued to occupy an uncertain position as a novel species of professional profile. If, on the one hand, there were thrusts towards emancipating analytic tasks from clerical and low-status work by introducing the distinction between programmer and coder, on the other hand the local and crafty nature of programming practices was undeniable. Anyway, as Nathan Ensmenger (2010) points out, gender patterns still played a significant role in downplaying the programming profession, defining a rather clear correspondence between the marginal, clerical work and the employment of female coders. The question of the status and identity of computer programmers was quite pressing for the nascent information industry along with doubts about the training – a clear criterion of recruitment –

[8] According to Paul Ceruzzi (1989), electrical engineering and computer science dominated computing activities in different stages along 40 years. Between 1940 and the early 1950s, the technology of electronics supported the feasibility of Babbage's concepts of automatic computing machines, whereas, between 1955 and 1975, computer science promoted the digital approach into computing.

computer employees had to undergo in order to adequately face the various challenges the new machines brought up.

By the mid-1950s, a new conception of programming as art developed and paved the way for the tensions between different computer programming cultures that would appear later. During this period, computer programmers had to deal with hardware limitations and mathematical analysis, which required several compromises between speed and accuracy of operation. Given these technical conditions, people at work on computers developed a peculiar type of expertise that combined analytic procedures and individual creativity. This blend of imagination and high precision led Frederick Brooks (1995) to juxtapose the figure of the programmer to that of the poet: the relationship between conceptual structures (the poet's imagination) and the code (the poet's words) are a matter of "magical incantation" and "perfection", more than of mechanical operations. During the 60s, the novel rhetoric surrounding computer programming solicited managers, such as Brooks, during the 1960s to rethink conventional management techniques in the light of the new organizational requirements demanded by the nascent computer profession[9]. What is most significant about the poetry metaphor is the unprecedented central role assigned to software programmers within the production process and the consequent effort for managers to couple industrial demand and the creativity of programmers/artists.

Despite the fact that programming was meant as a "black art" by industry and technical literature between the 1950s and 1960s, the conformation of expertise and job requirements still remained hazy, wavering from those who emphasized craft techniques and situated knowledge, and others who used these features to denigrate the job, highlighting, conversely, the intellectual and scientific sides of the emerging profession. As we are going to see, in fact, these opposite views became apparent shortly afterwards, with the development of computer science. What was evident, even at that time, is that gender patterns were taken on from both the positions in order to denigrate the counterpart.

[9] Brooks became manager of IBM Operating System/360 in 1964, which was the software part of the larger System/360, designed to perform the complete range of applications.

4. What is a computer programmer? Labor practices and rhetoric on professionalism

The main benefit of using gender as analytic category in the study of science and technology lies reasonably in the demand to look not only at quantitative data that document the scarce presence of women in computing fields, but rather to employ feminist analysis so as to highlight the process of "masculinization" of the profession. As Thomas J. Misa remarks: «surprisingly, not enough is known about how and when and why the gendered culture of computing emerged» (Misa 2010, 8). Thus, keeping masculinization and feminization – then framing gender relations as social constructions and dynamic processes – on the same footing means rather to problematize the overlap between a subset of hegemonic meanings and a broader set of potential meanings, an operation that Roy Jacques (1996) has termed "semantic eclipse"[10].

At the beginning of the 1950s, a pressing issue started to threaten the growth of commercial computer industry, namely the increasing shortage of computer programmers flagged all over (see Ensmenger 2010; Abbate 2012). The need to recruit new workforce inevitably brought to the fore very contentious questions about the forms of expertise and training, labor organization and cultural values surrounding the nascent profession of programming. At a closer look, we can see that professional requirements and technical standards set up to build a new organizational and technical profile (the computer programmer) are laden with gender assumptions regarding female labor and the identity of the profession under construction. For two decades until the early 1970s, indeed, the debate over programming methods involved a struggle among different meanings, metaphors and ideas to define the programmer's professional and social identity.

[10] As Jacques observes: «In a lunar eclipse, observes on Earth see a small body blocking the view of a much larger one. In a semantic eclipse, a relatively small subset of meanings comes to block sight of a broader set of potential meanings. For instance, when a mode of rationality normative to Western culture, masculine behavior, and the modern era is simply called rationality, the only category remaining for the reasoning of other cultures, women, and other historical periods is *ir*rationality or *non*rationality. Only a small area of the domain of rational behavior is visible; the rest is eclipsed by it» (1996: 159, emphasis in the original).

According to Janet Abbate (2012), two particular visions of the profession at the time – automatic programming and software engineering – are crucial to understand the process of masculinization under construction. The introduction of the first experimental compiler by Grace Murray Hopper, and the consequent debate over *automation* in computer programming, brought about two opposite views about the configuration of the professional identity[11]: there were those who believed programmers would have been deskilled (even replaced), and others who thought automation would empower programmers. Different, relevant groups – technicians, managers, users – held these positions according to their own, often conflicting, interests, which, in turn, were influenced by the sex of the worker. As Abbate suggests, indeed, what is important to highlight with regard to the automation debate at the time is the weight of gender and labor dynamics, so that, for instance, framing programming as a subprofession would have meant to associate it with female workers. Even the supposed software crisis occurred in the 1960s and the raising of the software engineering method take different meanings if we look at the conflicts among different ideas of programming as a profession. Both Abbate (2012) and Ensmenger (2001; 2010) provide enough arguments to sustain that:

> Insofar as the software crisis existed at all, it was neither a distinct event nor a coherent description of prevailing conditions in the industry. It may be better viewed as an all-purpose complaint that reflected inflated expectations, labor tensions, and gendered assumptions about who could do programming and how they should behave. The crisis rhetoric also provided a rationale for those who wanted to change the direction of programming. (Abbate 2012, 96)
>
> Like the "worldwide shortage of information technology workers" of the current era, the "acute shortage of programmers" of the 1960s was about more than a mere disparity between supply and demand. The problem was not so much a lack of

[11] The first modern automatic programming system, called A-0 and written by Hopper at UNIVAC, allowed to considerably reduce the time of execution of a program in a computer machine. Technically, the compiler took advantage and improved those reusable portions of code – called subroutines – by translating them into a program in machine code. This notable progress paved the way for the development of the first widely general- purpose programming language, FORTRAN, developed at IBM between 1954 and 1957 (see, Ensmenger 2010; Abbate, 2012).

> computer specialists per se but rather the lack of a certain kind of computer specialist. Teasing apart just what that certain kind of specialist was supposed to be goes a long way toward understanding the larger social and political context of these debates. (Ensmenger 2001, 70)

These two observations are rather remarkable, not only because they call into question the reasons behind the crisis, going beyond the acknowledgment of numbers, but also because they solicit one to take on the question of labor shortage and recruitment in IT industry nowadays under different terms.

In 1968, the North Atlantic Treaty Organization (NATO) sponsored an international conference where software programming was to be discussed. In that venue, various communities were invited to put forward their own concerns on the issue. According to Sandy Payette (2014), the divarication between two different interpretations about the status of the profession can be embodied by the figures of Grace Murray Hopper and Edsger Wybe Dijkstra. The former is famous for having written the first computer compiler at Remington Rand, whereas the latter is known for having won the Turing Award and set up computer science as an academic discipline. Their training, intellectual background and ideas clearly marked out different attitudes to computer programming and beliefs about its future. If Hopper held up software programming as an applied knowledge, connoted by pragmatism, urgency, opportunity, and collaboration, Dijkstra believed in a change of paradigm so that he strongly concurred to reframe programming as "software engineering", recognizing it as an "intellectual challenge" (quoted in Payette 2014, 67), with the same dignity of art and science. Such positions, far from just denoting two different assessments and uses of the machine, outline divergent economic interests, labor organization and practices as well as public acknowledgement and prestige, all aspects which are, consciously or not, biased by gender stereotypes. The first clear evidence of this is the absence of women at the NATO conference, notwithstanding their prominent role in the early computing era both as practitioners and knowing subjects; the absence of Hopper herself at a conference sponsored by NATO is quite meaningful, as Payette suggests, given her leading role in the US Navy. Moreover, following the various works on the relationship between gender and language (see Wodak 1997; Walsh 2001; Weatherall 2002), it is fundamental to investigate how discursive practices are mobilized in order to produce and reproduce gender configurations on the one hand, and how gender constructs are functional to build a cultural/technical hegemony by relevant social groups on the other[12]. With regard to the co-construction of gender relations and computer programming, there is a passage about Hopper's work from a speech by Dijkstra, which is extremely eloquent:

> Captain Hopper spoke officially about "Programming Languages"; her *[sic]* real subject was how she had acted as midwife to COBOL and she talked more about the Pentagon and the U.S. Navy than about programming. (Quoted in Payette 2014, 2014, emphasis in original)

The attempt to criticize, if not to downplay, Hopper's contribution to programming through a gendered language is rather apparent insofar as it recalls the classic division between productive and reproductive labor illustrated by feminist and gender analysis on labor (see, among others, Schwartz Cowan 1983; Piccone Stella and Saraceno 1996).

Furthermore, as Abbate points out (2012), the rise of software engineering sanctioned at the Garmisch conference is to be intended more as a change in guise than a substantial reform of computer programming. The term 'engineering', indeed, was brought up to sustain a novel structure whose components (abstraction, modularity and conditional loops), in reality, predated it. The reference by Abbate to the "unspoken ideas about which gender could best elevate the practice and status of programming" (2012, 103) echoes in many respects the attempt by Misa (2010) to frame the gender gap in computing by arguing in favor of gender biases encoded in professional culture rather than formal discriminations. That the high presence of women – although in deskilled labor positions - was intended as a temporary opportunity is not only demonstrated with reference to wartime, but also during the economic crises, as

[12] With the term 'cultural hegemony', I refer here to the intellectual thinking of Antonio Gramsci (1948/2007), further developed by Stuart Hall (1987) within Cultural Studies. Unlike 'domination' and 'ideology', hegemony needs consensus from popular groups to be effective, its power is temporary and constructed through the intervention of social, political, economic, cultural and media structures.

emerged from Marie Hicks' analysis (2010) on computerization in the British public service. After the economic troubles and labor shortage in high-tech industry in the mid 1960s, which favored a nearly equal treatment for women and men entering computing workforce, young female computer operators resigned due to low payments and bad job conditions. The Royal Air Force, then, decided to hire middle-aged married woman as machine operators since this kind of job profile was supposed to not have career ambitions and particular economic demands; thus, women were perceived not to be the best candidates for computing careers. On the other hand, private companies hired men as machine operators "because they offer a complete career to such people, and partly, as was said earlier, because it is felt that the computer field is generally a young man's domain. […] The young man seems to represent the 'best bet' if career opportunities and financial rewards are satisfactory" (quoted in Hicks 2010, 108). Therefore, technical skills were not a matter of concern for hiring women; the issue at stake regards, instead, the social expectations about female workforce, considered as temporary and not suitable for professional careers within computing industry. Not "the best bet", indeed.

5. Conclusion

In discussing the issue of 'heterogeneity' related to the processes of stabilization and standardization, Susan Leigh Star (1991) keenly remarks how power belongs to those able to impose the metaphors that shape the worlds we live in. Accordingly, reading the historical circumstances related to the birth of computing as a professional expertise and labor organization through feminist scholarship (see Faulkner 2001) means not just to highlight the contribution of women to technological development, but rather to shed light on the gendered nature of interests, beliefs and rhetoric upheld by different communities of practices (Lave and Wenger 1991) in computer work.

In this article, thus, I have attempted to provide a picture of the nascent digital computer fields as not just a matter of inventors and individual enterprises, but rather as a tangle of social and political assumptions on labor, gender and technology. I have started by presenting the introduction of the ENIAC through the stories of women at work on software components. Even at the time, in fact, the hard/soft split, well-articulated by Paul Edwards (1990), was an ideological construct useful to draw a demarcation between the analytical thinking embodied by male designers at work on hardware, and the mechanical work carried out by female operators. In fact, a closer look at the nature of work the ENIAC girls undertook demonstrates how the constructed image of the computer machine as divided into hardware and software components, although inadequate even from a technical point of view, served as a good argument to shape the configuration of the computer professions.

Between the 1950s and the 1960s, a novel metaphor of computer programming as a "black art" came out to improve programmers' labor conditions and general reputation. It concurred to define quite clear boundaries and tensions among programmers, computer scientists and managers, becoming evident with the recruitment practices (aptitude tests and psychological profiles) developed to face the software crisis emerged at the end of the 1950s on the one hand, and during the conference sponsored by NATO in Garmisch on the other. This venue gave birth to the software engineering approach, which, as explained in the third paragraph, was, more than other things, an attempt to gloss the image of the programming profession without getting through real changes. As emerged from the challenge between the growing computer science and an applied approach to computing, the attempt to strengthen the influence of scientific thinking was marked out by clear conceptions about the gendered division of labor, which assigned women to domestic duties rather than to professional careers.

The aim to frame the issue of 'professionalism' in computer industry within feminist perspectives on science and technology also speaks to the solicitation (Haigh, 2015) to provide further analyses that put the history of computing in conversation with the history of computer science. As compelling case studies have pointed out (see Lerman *et al.* 2003), in fact, the use of gender as an analytical tool – and not as a mere variable – in the study of science and technology invites scholars and researchers to interrogate not only the boundaries between men and women, masculinization and feminization, but also those dichotomies, such as hardware/software, practice/knowledge, skilled/unskilled, user/producer, which have long regulated – and still do – discursive and material practices surrounding technology.

References

Abbate, J. (2012), *Recoding Gender. Women's Changing Participation in Computing,* Cambridge, The MIT Press.

Bijker, W. E. (1987), “The Social Construction of Bakelite: Toward a Theory of Invention. New Directions in the Sociology and History of Technology”, in W. E. Bijker, T. P. Hughes, and T. Pinch (eds.), *The Social Construction of Technological Systems,* Cambridge, The MIT Press, pp. 159-187.

Brooks, Jr., F. P. (1995), *The Mythical Man-Month: Essays On Software Engineering,* Boston, Addison-Wesley.

Ceruzzi, P. (1989), *Electronics Technology and Computer Science, 1940-1975: A Coevolution*, in <<IEEE Annals of the HIstory of Computing >>, vol. 10, no. 4, pp. 257-275.

Edwards, P. N. (1990), *The Army and the Microworld: Computers and the Politics of Gender Identity*, in <<Signs>>, vol. 16, no. 1, pp. 102-127.

Ensmenger, N. L. (2001), *The 'Question of Professionalism' in the Computer Fields*, in <<IEEE Annals of the History of Computing >>, vol. 23, no. 4, pp. 56-74.

Ensmenger, N. (2010), *The computer boys take over,* Cambridge, MIT Press.

Faulkner, W. (2001), *The Technology Question in Feminism: A View from Feminist Technology Studies,* in <<Women's Studies International Forum>>, vol. 24, no. 1, pp. 79-95.

Fritz, W. (1996), *The Women of ENIAC*, in <<IEEE Annals of the History of Computing>>, vol. 18, no. 3, pp. 13-28.

Gramsci, A. (1948), *Gli intellettuali e l'organizzazione della cultura,* Torino, Einaudi.

Grier, D. A. (2005), *When Computers Were Human,* Princeton, Princeton University Press.

Haigh, T. (2015), *The Tears of Donald Knuth*, in <<Communications of the ACM>>, vol. 58, no. 1, pp. 40-44.

Hall, S. (1987), *Gramsci and Us*, in <<Marxism Today>>, pp. 16-21.

Hicks, M. (2010), “Meritocracy and Feminizatoin in Conflict. Computerization in the British Government”, in T. J. Misa (ed.), *Gender Codes. Why Women are Leaving Computing,* Hoboken, Wiley-IEEE Computer Society, pp. 95-114.

Jacques, R. (1996), *Manufacturing the Employee: Management Knowledge form the 19th to 21st Centuries,* London, Sage.

Kline, R. (2006), *Cybernetics, Management Science, and Technology Policy: The Emergence of "Information Technology" as a Keyword, 1948-1985*, in <<Technology and Culture>> vol. 47, no. 3, pp. 513-535.

Lave, J., Wenger, E. (1991), *Situated Learning. Legitimate Peripheral Participation,* Cambridge, Cambridge University Press.

Leigh Star, S. (1991), "Power, technologies and the phenomenology of conventions: on being allergic to onions", in J. Law (ed.), *A Sociology of Monsters: Essays on Power, Technology and Domination*, London: Routledge, pp. 26-56.

Lerman, N. E., Oldenziel, R., Mohun, A. P. (eds.) (2003), *Gender & Technology. A Reader,* Baltimore, John Hopkins University Press.

Light, J. S. (1999), *When Computers Were Women*, in <<Technology and Culture>>, vol. 40, no. 3, pp. 455-483.

Margolis, J., Fisher, A. (2002), *Unlocking the Clubhouse. Women in Computing,* Cambridge, The MIT Press.

Misa, T. J. (eds) (2010), *Gender Codes. Why Women Are Leaving Computing,* Hoboken, Wiley-IEEE Computer Society.

Owens, L. (1986), *Vannevar Bush and the Differential Analyzer: the Text and Context of an Early Computer*, in <<Technology and Culture>> , vol. 27, no. 1, pp. 63-95.

Payette, S. (2014), *Hopper and Dijkstra: Crisis, Revolution, and the Future of Programming*, in <<IEEE Annals of the HIstory of Computing>>, vol. 36, no. 4, pp. 64-73.

Piccone Stella, S., Saraceno, C. (eds.) (1996), *Genere. La costruzione sociale del femminile e del maschile,* Bologna, Il Mulino.

Pinch, T. J. (1992), *Opening Black Boxes: Sciences, Technology and Society*, in <<Social Studies of Science>>, vol. 22, no. 3, pp. 487-510.

Pinch, T. J., & Bijker, W. E. (1984), *The Social Construction of Facts and Artifacts: Or How the Sociology of Science and the Sociology of Technology Mights Benefit Each Other*, in <<Social Studies of Science>>, vol. 14, no. 3, pp. 399-441.

Schwartz Cowan, R. (1983), *More Work for Mother: The Ironies of Household Technology from the Open Hearth to the Microwave,* New York, Basic Books.

Walsh, C. (2001), *Gender and Discourse: Language and Power in Politics, the Church and Organizations,* London, Longman.

Weatherall, A. (2002), *Gender, Language and Discourse,* London, Routledge.

Winner, L. (1980), *Do Artifacts Have Politics?,* in <<Daedalus>>, vol. 109, no. 1, pp. 121-136.

Wodak, R. (1997), *Gender and Discourse,* London, Sage.