Different aspects of bacterial network communication english language essay

Linguistics, English

egin{abstract}In this paper we discuss the different aspects of bacterial network communication. The potentialities of designing routing network schemes based on bacteria motility will depend on the genes coding for the intracellular and intercellular communication molecular devices. An additional element is given by the " mobilome" which is related to horizontal gene transfer. First, by using a multi-objective optimization procedure, we search for the optimal trade off between energy production, which is a requirement for the motility, and the biomass growth, which is related to the overall survival and fitness of the bacterium. We made use of flux balance analysis of genome-scale biochemical network of Escherichia coli k-13 MG1655. Then, as a second case study we analyse energy and biomass properties of the bacterium Geobacter sulfurreducens. This bacterium is usually grown on a graphite electrode, providing acetate as food and quickly build an electrically dynamic biofilm matrix where electron transfer takes place across the nanowires. Geobacter species produce higher current densities than any other known organism in microbial fuel cell. Recent experiments have shown that Geobacter's electric nanowire has large evolutionary potential (a new evolved strain has eight times more electric current across the electrode than the original strain). Our methodology could estimate the evolutionary potential and help in designing optimal multi scale properties from networks to behavior. end{abstract}% section{Introduction} % no IEEEPARstartsection*{Communications in bacteria}Recently a simulation program, called the WholeCell (http://wholecell. stanford. edu; ~cite{Karr}) provided a fine description of many life cycle processes of a small bacterium (the genome is about 1/10th of that of Escherichia coli)

emph{Mycoplasma genitalium}: metabolism, replication of the genome, and cell division (9 hours or about 32, 000 repetitions of the simulation loop). The WholeCell model is composed by 28 distinct modules and includes a metabolic model of 441 chemical reactions. This model is analysed using flux balance analysis (FBA) that provide the solutions (reactions concentrations) satisfying the optimisation of the most efficient use of available resources, such as nutrients. Organic compounds are converted to carbon skeletons for the synthesis of various cell components and for the production of energy. This is possible through the regulation of the reactions of anabolism and catabolism. Bacteria could be used to build nano communication networks that operate in microfluidic devices, body area networks or other cite{LioSasi}. The fitness of a bacterium is particularly concentrated on the speed of dividing, but the division time depends on reaching a certain biomass. In order to achieve a biomass, the bacterium needs to locate a source of food and move towards it. We believe that the bacterium behavior could be analysed using multi-optimisation techniques ~cite{Costanza}. The results of the multi-objective optimization is not a single solution (such as in a single optimization problem), but a set of non-dominated points, which form the Pareto surface, i. e. the best trade-off design for the set of pathways considered, leading to optimize simultaneously multiple cellular functions of interest. For each Pareto optimal solution, it will be possible to compute the robustness, the sensitivity and identifiability integrated with all the available data. The information on sensitivity (the elements that have a large influence on the system outputs are considered sensitive) of biological networks and pathways could be used to suggest where and how much to

modify a metabolic network. We assume that the energy for the location of food source is in high demand and has higher priority than the biomass growth and division. section*{The coupling between metabolism and the chemotaxis device}A bacterium typically swims by alternating straight runs with short periods of tumbles that randomly reorientate the next run. Motile bacteria suppress tumbles when they head either up concentration gradients of attractants ordown gradients of repellents. Motile bacteria synthesize proteins for chemotaxis including flagellaformation when the substrate concentration, i. e. food, becomes low. The synthesis and function of the flagellar and chemotaxis systemrequires the expression of a network of more than 50 genes, therefore it is genomically and metabolic expensive. Using the proteinscoded by these genes, a bacterium use receptors to sense the spatial gradient and compares the instantaneous concentrationof carbon sources. Although existing models of bacterial chemotaxis do not take into account the metabolism, i. e. responding to a gradient of molecules, it is known that metabolism modulates chemotaxis and motility behavior. According to this, in this paper we study the metabolism as a trade -off between energy (required for motility) and the biomass (required for the growth). A decrease in biomass due to starvation would require spending resources towards searching new source of food and therefore accomplishing chemotaxis specific signal transduction, through the direct modulation of flagellar rotation. section*{Innovation in communication: the mobilome}Bacterial conjugation is a genetic transfer that involves cell-to-cell between donor and recipient cells. The importance of horizontal/lateral gene transfer (LGT) in shaping the genomes of prokaryotic organisms has been

recognized in recent years as a result of analysis of the increasing number of available genome sequences. LGT is largely due to the transfer and recombination activities of mobile genetic elements. In bacteria, horizontal gene transfer is often mediated by conjugative genetic elements that transfer directly from cell to cell. Integrative and conjugative elements, also known as conjugative transposons, are mobile genetic elements that reside within a host genome but can excise to form a circle and transfer by conjugation to recipient cells. Gene duplication has long been recognised as an importantmechanism for the creation of new gene functions cite{massingham2001analysing}.egin{figure}centeringincludegraphics[scale= 0. 4]{duplication2. pdf}caption{In A, the tree describes the events of the duplication or LTG. Recent events are separated by short distances (M is the number of mutations changing the protein functions) in the y axis. The symbols refer to genes with similar functions, for example sensors, regulatory or flagellar proteins. In B, the LTG events between three bacteria (B1-3) generate similar but not the exactly same metabolic networks; in green are the sensor or flagellar proteins; in red the protein interactions.}label{fig: duplication}end{figure}Formally, let y be the array representing the sequence of the L genes of the organism. During the evolution process, a gene or a subsequence of genes (e. g. an operon) can be duplicated and inserted in the sequence. Without loss of generality, let us assume that the last k genes are duplicated:$$ y=(y_1,..., y_L) quad longrightarrow quad y=(y_1,..., y_L, y_{L+1},..., y_{L+k}).$$This process is called gene amplification or gene duplication. It has been estimated that 50\\% of Escherichia coli genes are paralogs i. e. have arisen from a gene

duplication event, as opposed to orthologs, which have arisen due to species divergence. The duplicated gene or the gene from a LTG event may code for a sensor signal. Initially, the following {it condition of duplication} holds: $ y_l= y_{l-k}, forall l= L+1,..., L+k$. In fact, since the duplication is a stochastic process, the condition of duplication is not always guaranteed. However, after the duplication, mutations occur on new and existing genes, thus we obtain the final string $ y=(y_1,..., y_{L'})$, where $L'= L+k$. Let us suppose that the gene y_L was responsible for the reaction $D_i ightarrow D_j + H_r$, and therefore for the instruction $inc(i, r, j)$ in the Minsky's register machine (RM, formally defined in Section ef{sec: rm}).%Let us consider the {it Minsky's register machine} (RM), i. e. a finite state machine augmented with a finite number of registers.. After the duplication, both y_L and y_{L+k} will code for the same reaction $D_i ightarrow D_j + H_r$. Conversely, after the mutation, y_{L+k} will code for another reaction, say $D_{i'} ightarrow D_{j'} + H_{r'}$. As a result, the complexity of the metabolic machine has increased, since a new reaction $inc(i', r', j')$ is now operating in the RM. Starting from an ancestor, each duplication followed by a mutation shapes the computational capability of the metabolic machines represented by its metabolism. The mutation is a stochastic process that creates the possibility of a new instruction of the metabolic machine, while the natural selection can keep or discard this new instruction. Hence, the computational complexity of an organism evolves on the basis of stochastic processes and natural selection. The genome amplification allows the organism to increase the range of chemical reactions available, increasing also the range of increment and decrement instruction in the RM

associated with the metabolism. As a result, the metabolic machine increases its computational power.egin{figure*}[! t]centeringincludegraphics[scale= 0. 55]{communication. pdf}caption{The communication between two bacteria. Communication between genes(top and right pictures) is useful if one bacterium needs pieces of genome fromanother bacterium in order to engulf them and increase its computationalcapability (left). This process allows a genome enriched with respect to theobjectives that the bacterium aims at optimizing (bottom). In a later step, theenriched bacterium may communicate with the bacterium on the right sendingother DNA fragments.}label{gene_engulfment}end{figure*}section{Communication and Turing machines}Let us now turn into the relation between computation and metabolism inspired by Turing cite{turing1952chemical}. Turing states that an organism, most of the time, develops from one pattern into another. %In order to follow this general process without simplifying assumptions (which are often made when conducting theoretical analyses), one needs not only to have a theoretical approach, but also to treat particular cases in detail with the aid of a digital computer. Many years later, Bray cite{bray1995protein} argued that a single protein is able to transform one or multiple input signals into an output signal, thus it can be viewed as a computational or information carrying element.%The cellular behaviour, controlled by complex regulatory circuits, strongly depends on the signal transduction proteins, which integrate multiple input signals responding simultaneously to all of them cite{lim2002modular}. Landweber and Kari cite{landweber2003universal} provided a model to view the unscrambling of

genes in ciliates as a computational process and this has much in common with the Adleman's algorithm cite{adleman1994molecular} in graph theory. Hence, a guided genome recombination system can simulate a Turing Machine (TM), and therefore a functional macronuclear gene can be viewed as the output of a computation carried out on the micronuclear sequence cite{amos2004cellular}. Following this line of thought, we provide a framework to show that bacteria could have computational capability and act as molecular machines. This relationship is based on the mapping between the metabolism and a RM (equivalent to a Turing Machine, TM). Specifically, we think the reactions in the bacterium as increment/decrement instructions of the RM, where the RM registers count the number of molecules of each metabolite.%Remarkably, as the biological system grows larger, reaching the desired multiple input/output performance becomes a difficult task, thus some sort of machine optimisation is required. To this end, we provide a novel algorithm called Genetic Design through Multi-Objective optimisation (GDMO), with the aim of programming molecular machines to maximise the yield of a desired metabolite.%By giving to a bacterium a certain amount of input metabolites, its metabolism follows a given pattern and produces output metabolites. Therefore, by changing these input quantities, one could force it to move to another development pattern cite{turing1952chemical}. It is well known that a von Neumann architecture is composed of a processing unit, a control unit, a memory to store both data and instructions, and input-output mechanisms.%The bacterium takes as input chemicals (substrates) necessary for his growth and duplication, and through its biochemical network (coded by the genes of its genome), produces

metabolites as output. We propose an effective formalism to map the von Neumann architecture to an entire bacterial cell, which becomes a molecular machine. %In particular, the genome sequence is thought of as an executable code specified by a set of commands in a sort of ad-hoc low-level programming language.%Each combination of genes is coded as a string of bits $y in left { 0 , 1 ight }^L$, each of which represents a gene set. Turning off a gene set means turning off the chemical reactions associated with it. %Each bit in y is a gene set that distinguishes between single and multi-functional enzymes, isozymes, enzyme complexes, enzyme subunits.

%

%

%The string y acts as a program stored in the memory unit.%The memory unit contains the string y, which is a program written in an ad-hoc low-level programming language. % Proposta di Pietro%The control unit is a function g_{Phi} that defines a partition of the string, and is uniquely determined by the pathway-based clustering of the chemical reaction network. %The function g_{Phi} interprets the binary string y and knocks gene sets out, thus turning syntax into semantics.

%

We model the processing unit of the bacterium as the collection of all its chemical reactions, so as to associate the chemical reaction network of bacteria with a TM cite{EasyChair13}. %cite{soloveichik2008computation} %Let us consider the {it Minsky's register machine} (RM), i. e. a finite state machine augmented with a finite number of registers. The Minsky's RM has

been proven to be equivalent to the TM. In order to map the chemical reaction network to the RM, we define:%cite{soloveichik2008computation}% (i) the set of state species ${D_i}$, where each D_i is associated with the state i of the RM; (ii) the set of register species ${H_r}$, where each H_r is associated with the register r of the RM, and therefore represents the molecular count of species r. The instruction $inc(i, r, j)$ can be viewed as the chemical reaction $D_i ightarrow D_j + H_r$. The instruction $dec(i, r, j, k)$ can be viewed as either $D_i + H_r ightarrow D_j$ or $D_i ightarrow D_k$ depending on whether $H_r> 0$ or $H_r= 0$ respectively. In our FBA approach, the variables are the fluxes of the reactions in the network, therefore a high flux corresponds to both a high rate of reaction and a high mass of products. Consequently, given the increment reaction $inc(i, r, j)$, the value of H_r is positively correlated with the reaction flux; conversely, in the decrement reaction $dec(i, r, j, k)$, when $H_r> 0$ the value of H_r is negatively correlated with the reaction flux.

%

%By investigating the whole metabolism of bacteria considering pathways of many proteins, we extend the above mentioned Bray's idea, i. e. thinking of a protein as a computational element. Likewise, the cell as a computational element receives, processes, and responds to inputs from the environment. Here we use {sc GDMO} cite{costanza2012robust} to obtain Pareto fronts representing multi-objective optimisations in the metabolism. Each point of the Pareto front provided by {sc GDMO} is a molecular machine to execute a particular task. Pareto optimality allows to obtain not only a wide range of

Pareto optimal solutions, but also the {it best trade-off design}. In Figure ef{fig: pareto_ac_succ} we show a Pareto front obtained with {sc GDMO} when optimising acetate and succinate.

%

%Finally, we propose a solution for the problem of making the sensitivity analysis pathway-dedicated: we develop the Pathway oriented Sensitivity Analysis ({sc PoSA}) to investigate the functional components of the molecular machine and detect the most sensitive ones.%The robustness analysis supports GDMO and PoSA in that it indicates the components of the molecular machine that are likely to ``fail''.% To sum up, are bacteria unconventional computing architectures? Our work suggests we may answer in the affirmative. Optimal genetic interventions in cells, framed as optimal programs to be run in a molecular machine, can be exploited to extend and modify the behaviour of cells and cell aggregates. For instance, programs can instruct cells to make logic decisions according to environmental factors, current cell state, or a specific user-imposed aim, with reliable and reproducible results.%This would lead to effectively modifying and harnessing biological organisms for our purposes. section*{Communication between bacteria driven by Pareto fronts}A Pareto front is the result of an optimization technique needed when a systema given phenotype cannot be optimal at all the tasks it performs, andparticularly when tasks are in contrast with each other. Two communicating organisms can harness the many-objective Pareto optimality tomake a trade-off decision that allows to define their behavior. The Paretofront allows to maximize or minimize two or

more target metabolites in anorganism, thus obtaining new optimal strains specialized in many aimssimultaneously. By adopting a trade-off strategy, an organism is able to optimizesimultaneously several biotechnological targets, e. g. the input and the outputof the computation he carries out. Given r objective functions $f_1,..., f_r$ to optimize, the problem ofoptimizing in a multi-objective fashion can be formalized as$$smashdisplaystyle{max_{x}} (f_1(x), f_2(x),..., f_r(x))^intercal,$$% $max_{x}(f_{1}left(x ight), f_{2}(x), ldots , f_{r}(x))^{intercal}, $% $smash{displaystylemax_{x}} (f_1(x), f_2(x),..., f_r(x))^intercal$where x is the variable in the search space. Without loss of generality, in thedefinition all the functions are maximized (however, minimizing a functionf_i is equivalent to maximizing $-f_i$). The output of a multi-objective routine is a set of Pareto optimal points, which constitute the emph{Pareto front}. A solution y^* is Pareto optimal ifthere does not exist a point y such that $f(y)$ dominates $f(y^*)$, i. e.$f_i(y)> f_i(y^*), forall i= 1,..., r$, where f is the vector of r objectivefunctions that have to be maximized in the objective space. A bacterium can use the Pareto-front to find the best trade-off between two ormore requirements. Since the bacterium has always more than one functions, thedecision whether to communicate with another bacterium has to take into accountthe output of an internal multi-objective optimization routine. For instance, an {it Escherichia coli} whose objectives are the production of acetate andbiomass, obtains the Pareto front in Figure ef{ecoli_front}. (the model takeninto acccount is the {it E. coli} by Orth et al. cite{orth2011comprehensive}.) A Pareto front produced by an organism is the set of all the phenotypes thatovercome all the feasible phenotypes

dominated on all taskscite{shoval2012evolutionary}. Communicating with other bacteria is intended toincrease the computational capability of the bacterium, and therefore moves thePareto front towards the best unfeasible point, which is located at the topright of the acetate-biomass graph. Nevertheless, this can decrease thecapability of the bacterium to produce a third-objective, and therefore thedecision may require a three-objective optimization routine. The communication happens when two bacteria share DNA fragments (see Figure ef{gene_engulfment}).egin{figure}centeringincludegraphics[scale= 0. 5] {atp_biomass. pdf}caption{Pareto front of the energy (ATP) (y- axis) versus biomass (x-axis) in E. coli; the x axis represents the different strains; a more ``energetic'' strain would represent a choice towards an increased motility; the biomass choice would represent a choice towards a faster replicating strain.}label{fig: atp_biomass}end{figure}egin{figure}[! t]centeringincludegraphics[scale= 0. 6]{pareto_ac_succ}caption{Pareto fronts for the simultaneous maximisation of succinate and acetate production obtained by {sc GDMO} in anaerobic and aerobic conditions ($O_2 = 10 mmolh^{-1} gDW^{-1}$), with glucose feed equal to $10 mmolh^{-1} gDW^{-1}$. The acetate represents sources of energy; the succinate enters the krebs cycle.} %In all our experiments biomass is constrained to assume values greater than or equal to $0. 05 mmolh^{-1} gDW^{-1}$, in order to guarantee a minimal growth rate.}label{fig: pareto_ac_succ}end{figure}

%

egin{figure}[! t]centeringincludegraphics[scale= 0. 65]{ecoli2. pdf}caption{Result of the two-objective optimization routine carried out on the{it E. coli} model Since the communication among bacteria allows to share DNAfragments, ant therefore increases their computational capabilities towards oneor more objectives (e. g., acetate and biomass) in which a bacteriumspecializes.}label{ecoli_front}end{figure}section{Metabolic networks as vehicles for communication}label{sec: rm}Inspired by Brent and Bruck cite{brent20062020}, who studied similarities and differences between biological systems and von Neumann computers, we propose a mapping between the von Neumann architecture and bacteria. Specifically, the metabolism of a bacterium can be viewed as a Turing Machine.

%

The bacterium takes as input the substrates required for its growth and, thanks to its chemical reaction network, produces desired metabolites as output. The string y acts as a program stored in the RAM cite{EasyChair13}. Let us consider the multiset Y of the bits of y. A partition Pi of the multiset $ Y = {y_1, y_2,..., y_L} $ is a collection ${b_1, b_2,..., b_p}$ of submultisets of Y that are nonempty, disjoint, and whose union equals Y. The elements ${b_s}_{s= 1,..., p}$ of a partition are called blocks. We denote by $P(Y; p)$ the set of all partitions of Y with p blocks. $P(Y; p)$ has a cardinality equal to the Stirling number, namely $left| P(Y; p) ight| = S_{L, p} $.% and by $P(y)$ the set of all partitions of y by $P(y)$. In order to formalise the control unit behaviour, let us define the

function:$$ g_{Phi} : {0, 1}^L longrightarrow igcuplimits_{y in {0, 1}^L } P(Y; p), qquad ar{y} in {0, 1}^L longmapsto Pi in P(ar{Y}; p), $$oindent where the partition Pi is uniquely determined by the pathway-based clustering of the chemical reaction network. We can formalise this clustering as a p-blocks partition Phi of the set of the bit indexes in the string y. In particular, if we denote by $left[L ight]$ the set of the first L natural numbers, we have $Phi in P(left[L ight]; p)$ cite{EasyChair13}. The partition Phi allows the control function g_{Phi} to partition the multiset Y associated with the string y.% (see Figure ef{fig: partition}). The function g_{Phi} turns syntax into semantics cite{EasyChair13}, i. e. is control unit that translates the binary string y and employs it to turn gene sets on and off, considering also the pathways in the metabolism. Each element of the partition Pi is the submultiset b_s of all the gene sets related to reactions in the s-th pathway. The processing unit of the bacterium could be modelled as the collection of all its chemical reactions. Therefore, the chemical reaction network of bacteria can be associated with a TM cite{soloveichik2008computation}. Let us consider the {it Minsky's register machine}, i. e. a finite state machine augmented with a finite number of registers. Formally, a Minsky machine $mathcal{M}=(D, i_0, i_1, varphi)$ is composed of a finite set D of states, a finite set $H={H_r}_r$ of registers, and a multivalued mapping $varphi : D ackslash {i_0} longrightarrow { (H_r, i),(H_r, j, k) | H_r in H, j, k in D } $. The set D has two distinguished elements $i_0, i_1 in D$ representing the initial state and the halting state respectively. Each register H_r of the RM stores a non-negative integer. The instruction $inc(i, r, j)$ increments register r by 1 and causes the

machine to move from state i to state j through the mapping $varphi(i)= j$. Conversely, the instruction $dec(i, r, j, k)$, given that $H_r > 0$, decrements register r by 1 and causes the machine to move from state i to state j ($varphi(i)= j$); if $H_r = 0$, the machine moves from state i to state k ($varphi(i)= k$). The Minsky's RM has been proven to be equivalent to the TM cite{minsky1967computation}. Indeed, a RM is a multitape TM with the tapes restricted to act like simple registers (i. e. ``counters''). A register is represented by a left-handed tape that can hold only positive integers by writing stacks of marks on the tape; a blank tape represents the count `0'. The chemical reaction network of a bacterium can be mapped to the RM by defining cite{soloveichik2008computation}: (i) the set of state species ${D_i}$, where each D_i is associated with the state i of the RM; (ii) the set of register species ${H_r}$, where each H_r is associated with the register r of the RM, and therefore represents the molecular count of species r. The instruction $inc(i, r, j)$ represents the chemical reaction $D_i ightarrow D_j + H_r$, while the instruction $dec(i, r, j, k)$ represents either $D_i + H_r ightarrow D_j$ or $D_i ightarrow D_k$ depending on whether $H_r> 0$ or $H_r= 0$ respectively. The molecular machine performs the ``test for zero'' by executing the reaction $D_i ightarrow D_k$ only when H_r is over, since the r-th register cannot be decreased and the reaction $D_i + H_r ightarrow D_j$ cannot take place. In the FBA approach coupled with the metabolic machine, the variables are the fluxes of the chemical reactions, therefore a high flux corresponds to both a high rate of reaction and a high mass of products. Hence, given the increment reaction $inc(i, r, j)$, the value of H_r is positively correlated

with the reaction flux; conversely, in the decrement reaction $dec(i, r, j, k)$, when $H_r> 0$ the value of H_r is negatively correlated with the reaction flux. In a fixed volume V in which the reactions occur, given two reactions inc and dec with fluxes v_1 and v_2 respectively, the metabolism of the bacterium has a probability of error per step equal to $epsilon= v_2/(v_1/V+v_2)$. Since the simulated TM can be universal, the correspondence between metabolism and TM allows to perform any kind of computation through a set of species and chemical reactions characterised by their flux. As a result, bacteria can carry out at least any computation performed by a computer. A program embedded in a bacterium, whose metabolism works like a TM, could be able to implement the robust knockout strategy found by {sc GDMO} cite{costanza2012robust}.%egin{figure} %centering%includegraphics[height= 24mm, width= 61mm]{partition. pdf} %caption{The multiset Y associated with y is partitioned by Pi in p blocks. The elements of Pi are submultisets of Y, since y is a string of bits, thus 0 and 1 may occur more than once in the same subset. In this example, $Pi= left{ left{ y_4 ight}, left{ y_1, y_6, y_2 ight}, ldots, left{ y_5, ..., y_L ight} ight}$, $Phi= left{ left{ 4 ight}, left{ 1, 6, 2 ight}, ldots, left{ 5, ..., L ight} ight}$}%label{fig: partition} %end{figure}section*{The electric properties of Geobacter}The bacterium Geobacter sulfurreducens produces electricity from the organic matter due to electrically conductive pili. The pili of a population of this bacterium form an electric biofilm that provides a direct connection with the electrode surface. Geobacter species produce higher current densities than any other known organism. Geobacter is usually grown on a graphite electrode,

providing acetate as food and quickly build an electrically dynamic biofilm matrix. where electron transfer takes place across the nanowires. Our studies highlight the potentialities of choosing different strains according to the biomass - electronic properties (see figure ef{geo}). A larger biomass may turn into more dense biofilms while viceversa we could obtain better electronic properties of the biofilm.%egin{figure}[! t]egin{figure}centeringincludegraphics[scale= 0. 35, angle=-90] {geobacterpf. pdf}caption{Pareto front for the simultaneous maximisation of biomass and electron production of the Geobacter bacterium.}label{fig: geo}end{figure}

%

section{Conclusions}In this paper we have highlighted the links between bacterium communication, gene duplication, lateral gene transfer events and metabolic complexity. The methodology we propose allows to optimise simultaneously several objectives, i. e. the output of the metabolic ``computation'' versus communication carried out by bacteria. This approach highlights the complex behaviour that may arise in molecular machines; although the nano communication network and synthetic biology are still in their infancy, we foresee the potentialities to build and optimise synthetic organisms that could be designed for specific communication performances and networks.ibliographystyle{unsrt} {smallibliography{monacom_biblio}vfill%ibitem{IEEEhowto: kopka} %H.~Kopka and P.~W. Daly, emph{A Guide to LaTeX}, 3rd~ed. hskip 1em

plus% 0. 5em minus 0. 4em elax Harlow, England: Addison-Wesley,

1999.%end{thebibliography}% that's all folksend{document}