Matematica, Cultura e Società

Rivista dell'Unione Matematica Italiana

Silvia Benvenuti, Linda Pagli

Le scienziate dimenticate: Margaret Hamilton, le missioni sulla Luna, matematica e programmazione

Matematica, Cultura e Società. Rivista dell'Unione Matematica Italiana, Serie 1, Vol. **6** (2021), n.1, p. 23–35.

Unione Matematica Italiana

<http://www.bdim.eu/item?id=RUMI_2021_1_6_1_23_0>

Matematica, Cultura e Società. Rivista dell'Unione Matematica Italiana, Unione Matematica Italiana, 2021.

Matematica, Cultura e Società – Rivista dell'Unione Matematica Italiana
Serie I, Vol. 6, N. 1, aprile 2021, 23-35

Le scienziate dimenticate: Margaret Hamilton, le missioni sulla Luna, matematica e programmazione

Silvia Benvenuti
Università di Bologna
E-mail: silvia.benvenuti5@unibo.it

Linda Pagli
Università di Pisa
E-mail: linda.pagli@unipi.it

Sommario: *Guerra fredda, 1961: l'Unione Sovietica ha dato un terribile smacco agli Stati Uniti, mandando il primo uomo nello spazio. Il presidente Kennedy decide di tentare la carta della missione di conquista della Luna, difficilissima ma di grandissimo impatto mediatico.*

La giovane matematica Margaret Hamilton si trova quasi per caso a essere un personaggio chiave della missione: assunta come programmatrice quando la sua esperienza era minima, diventa in breve tempo tanto esperta da essere messa a capo del gruppo NASA responsabile del software del computer di guida dell'Apollo 11. Inventa nuovi metodi di coordinamento e controllo del software che diventeranno importantissimi nell'immediato futuro, pioniera del software engineering, *disciplina a cui lei stessa dà il nome.*

La complessità del programma Apollo 11 è enorme sia dal punto di vista ingegneristico che matematico e informatico, coinvolge centinaia di persone, può fallire in mille modi diversi, ma grazie anche alla scrupolosità di Margaret ha il successo che sappiamo. Il suo importantissimo contributo è premiato dal presidente Obama nel 2017.

Abstract: *Cold War, 1961: the Soviet Union gave the United States a terrible blow by sending the first man into space. President Kennedy decides to try the road of the mission to conquer the Moon, very difficult but of great media impact.*

The young mathematician Margaret Hamilton finds herself almost by accident to be a key figure in the mission: hired as a programmer when her experience was minimal, she quickly becomes expert enough to be put in charge of the NASA team responsible for the software of the guidance computer of the Apollo 11. She invents new methods of software coordination and control that will become very important in the near future, she was a pioneer of software engineering, a discipline to which she herself gives her name.

The complexity of the Apollo program is enormous both from an engineering, mathematical and computer science point of view, it involves hundreds of people, it can fail in a thousand different ways, but thanks to Margaret's scrupulousness it has the success we know. Her very important contribution is rewarded by President Obama.

1. – Introduzione

> I believe that this Nation should commit itself to achieving the goal, before this decade is out, of landing a man on the Moon and returning him safely to the Earth.

Lo afferma un temerario John Kennedy il 25 maggio del 1961 in un discorso al Congresso degli Stati Uniti, da cui storicamente ebbe inizio lo sforzo condotto da parte di URSS e USA per portare un equipaggio umano sulla Luna [7].

Fino a quel momento, missioni sovietiche e statunitensi senza equipaggio dirette alla Luna si erano alternate a un ritmo frenetico ma non per

Accettato: il 13 marzo 2021.

questo coronato da successo: dopo dieci fallimenti, equamente divisi tra americani e sovietici, il 14 settembre 1959 era toccato ai secondi centrare il primo allunaggio, centrando più propriamente con la loro sonda Luna 2 la superficie del satellite omonimo, per fracassarvisi a costituire il primo rottame terrestre depositato sulla Luna. Il 4 ottobre dello stesso anno Luna 3, sorvolando la faccia nascosta della Luna, mandava a Terra la prima immagine dell'emisfero fino ad allora sconosciuto, a competere con gli acquerelli di Galileo che dai tempi del *Sidereus Nuncius* raffigurano invece quello ormai ben noto. Seguirono, tra il 26 novembre 1959 e il 20 aprile 1964 quattordici ulteriori fallimenti, anche questi democraticamente distribuiti tra le due parti. L'URSS, però, aveva nel frattempo fatto il vero "colpo grosso": il 12 aprile 1961, infatti, aveva mandato il primo uomo in orbita, Jurij Gagarin, aggiudicandosi così il primato spaziale più ambito (il terzo consecutivo dopo quelli di Sputnik 1 e Sputnik 2, con a bordo la povera Laika). È forse per questo che Kennedy, poco più di un mese dopo, annunciò che entro un decennio i suoi astronauti avrebbero piantato la bandiera a stelle e strisce sulla Luna? La sfida pareva infatti estremamente temeraria, vista la lista degli insuccessi precedenti, che configuravano comunque un'indiscussa superiorità russa. Be', il punto è che Kennedy, all'indomani dello smacco subito per colpa di Gagarin, ben conscio dell'impatto mediatico della corsa allo spazio aveva chiaro che, in un modo o nell'altro, a questo spazio ci doveva arrivare. Gli scenari mediaticamente appetibili erano tre: costruire un laboratorio in orbita terrestre, spedire un uomo in orbita attorno alla Luna o farvelo addirittura atterrare. James Webb, allora amministratore della NASA, era ben certo che sui primi due obiettivi non c'era battaglia, i russi ci sarebbero arrivati senz'altro per primi; per non correre quindi il rischio di perdere un'altra battaglia, su suo consiglio Kennedy si focalizzò sull'ultimo, tra i tre di gran lunga il più oneroso, con un costo stimato di 22 miliardi di dollari nel corso del decennio.

Era peraltro un'epoca in cui il primato spaziale era un affare molto serio: dopo la seconda Guerra mondiale, le due superpotenze erano impegnate in un duro confronto muscolare che mirava (forse) a fare da deterrente a un ulteriore conflitto mondiale che nessuno aveva interesse a innescare. In questo senso, la conquista allo spazio era un'arma di intimidazione estremamente efficace: mettere in orbita un satellite era un messaggio in codice per comunicare alla potenza avversaria «ho un missile balistico intercontinentale, per ora ci trasporto un satellite, ma se voglio ci metto l'atomica, fai un po' tu». A capo del team missilistico statunitense c'era Wernher von Braun, ex nazista che aveva guidato il programma V2 che aveva terrorizzato l'Inghilterra e che, intuita la disfatta, si era consegnato con tutta la sua squadra all'esercito americano, che aveva pensato bene di farne il suo asso nella manica. Conduceva invece il programma avversario Sergej Korolëv il quale, dopo una dura reclusione e un successivo soggiorno forzato in Siberia ai tempi di Stalin, era stato individuato da Nikita Chruščëv come il personaggio chiave per guidare il loro progetto missilistico.

Come leggiamo in [7], «con il suo innocuo *bip bip* lo Sputnik rappresentava una plateale dimostrazione che i missili balistici intercontinentali sovietici potevano colpire gli Stati Uniti come e quando volevano». E il resto dell'iniziale corsa allo spazio non è altro che un *bip bip* sempre più forte.

D'altra parte, la corsa alla Luna presuppone il superamento di enormi difficoltà non solo di natura tecnologica, ma anche scientifica: capire come gestire il problema del trasferimento dalla Terra al suo satellite, garantendo al contempo agli astronauti coinvolti un ambiente di bordo nel quale fosse possibile sopravvivere, e un sicuro rientro a casa, pone questioni matematiche profonde e richiede una padronanza degli strumenti informatici a quel tempo niente affatto scontata. In questo articolo ci piacerebbe dare un'idea delle problematiche matematiche e informatiche soggiacenti, approfittando dell'occasione per raccontare il ruolo che in questa grande avventura, a fianco dei colleghi uomini di cui tutti conosciamo a memoria i nomi (e di cui ovviamente non vogliamo certo sminuire l'importanza), ebbe la ragazza che vedete nella foto di Figura 1, Margaret Hamilton, capo del progetto del software dell'Apollo Guidance Computer.

FIGURA 1 – Margaret Hamilton in una foto degli anni del programma Apollo.

2. – Margaret Hamilton

Nel 1960 Margaret arriva a Boston, con una laurea in matematica, per specializzarsi con un dottorato. Aveva un marito e una figlia piccola che portava sempre con sé. Dopo aver risposto a un annuncio del prestigioso Massachusetts Institute of Technology (MIT), ottiene un impiego temporaneo per sviluppare programmi per le previsioni meteo sui calcolatori LGP-30 e PDP-1 e rimanda il dottorato. In breve tempo diventa la programmatrice di riferimento al laboratorio e viene coinvolta nel progetto Apollo per l'invio di capsule sulla Luna di cui, in breve tempo, diventa la responsabile.

Nelle foto dell'epoca Margaret sembra una ragazzina con la sua minigonna, gli occhialoni e i capelli lunghi e sciolti molto anni '70; in realtà ha compiti importantissimi: per la missione Apollo 11, dirige il progetto del software dell'Apollo Guidance Computer, il computer di bordo del modulo di comando e del modulo lunare, cioè un computer da programmare per una missione critica e complessa, con il rischio di vite umane, avendo a disposizione strumenti e conoscenze molto limitate di quella disciplina [17].

Aveva già progettato il software per le missioni precedenti come l'Apollo 8, di cui racconta sempre un aneddoto: sua figlia di circa sei anni la stava osservando mentre testava il software degli astronauti sullo schermo. La bambina voleva giocare all'astronauta e, all'improvviso, il sistema si bloccò. Il problema era stato causato dal fatto che la figlia aveva premuto il tasto P01, che avviava un programma da eseguire prima del lancio e non durante il volo e il sistema non consentiva alle varie fasi di avere parti in comune. Margaret capì che la manovra involontaria della figlia aveva messo in luce un problema serio, che fece immediatamente presente ai suoi capi. Le fu risposto che gli astronauti erano persone molto ben preparate e non avrebbero mai e poi mai fatto errori del genere, le fasi diverse potevano restare distinte, perché non avrebbero interferito tra loro... e, come le ultime parole famose delle barzellette, fu puntualmente quello che accadde: il pulsante P01 fu premuto per sbaglio durante il volo da un astronauta, causando un arresto del computer di bordo. Il veicolo spaziale dovette essere pilotato dalla base e ci vollero ore per rimettere tutto a posto. Nelle missioni successive se ne tenne conto. Margaret era scrupolosissima, ossessionata dalle conseguenze dei possibili errori, passava notti insonni, si scontrava con gli ingegneri che spesso bloccavano le sue soluzioni dicendo che non si potevano realizzare, solo perché nessuno l'aveva mai fatto prima!

Per spiegare le difficoltà del progetto, che non era un semplice programma, ma un sistema di programmi da coordinare, sviluppati da persone diverse, e forse per essere meglio compresa dagli ingegneri dell'hardware, definì il lavoro complessivo come un progetto di *software engineering*, dando inizio a una nuova disciplina che in seguito ha mantenuto questo nome [6]. Gli ingegneri dell'hardware, nella loro progettazione, potevano basarsi su regole, quelli del software no, dovevano risolvere problemi mai risolti prima, con la difficoltà aggiuntiva di non poter essere dei principianti. Dovettero essere esperti fin dall'inizio e in qualche modo ci riuscirono. Inoltre le risorse del computer, spazio e velocità di calcolo, erano ai tempi molto limitate. Come se non bastasse poi il software doveva essere ultra affidabile, gli errori dovevano essere rilevati e corretti in tempo reale: c'era infatti in gioco la vita degli astronauti.

Margaret e i programmatori del suo gruppo realizzarono il sistema di programmi che era in grado di coordinare il flusso dei dati proveniente dal sistema di navigazione giroscopico, dal telescopio e da due radar, fornendo agli astronauti il controllo sui motori e su tutte le operazioni di bordo. Fu così possibile che il primo essere umano calpestasse la superficie lunare il 20 luglio del 1969. Ma proprio quando stavano per allunare si presentò un'altra emergenza: era comparso erroneamente sul display il segnale dell'attivazione del radar per il rientro della navicella. Il software progettato fu in grado di ignorare l'errore, che altrimenti avrebbe portato a un sovraccarico il computer di bordo, compromettendo le operazioni di sbarco. Situazioni analoghe a questa erano state previste da Margaret che aveva capito che era necessario organizzare i compiti del computer per priorità perché, in caso di conflitto di informazioni durante l'allunaggio, si attivassero le operazioni a priorità maggiore, escludendo quelle non necessarie.

Andiamo quindi a vedere un po' meglio che genere di problemi matematici pone una missione Terra – Luna con equipaggio, e perché era necessario un sistema di calcolo computerizzato per risolverli in modo efficiente.

3. – Matematica della Luna

Abbandonare l'atmosfera terrestre, entrare in orbita, atterrare sulla superficie della Luna e, possibilmente, rientrare sani e salvi a Terra a fine missione sono problemi piuttosto complicati dal punto di vista della matematica coinvolta. Sebbene affrontare queste questioni non fosse tanto compito di Margaret quanto piuttosto dei colleghi ingegneri che realizzavano il calcolo di funzioni direttamente implementate in hardware con delle ROM, dei colleghi matematici e di altri programmatori, riteniamo interessante in questo paragrafo dare un'idea almeno di una piccola parte degli strumenti matematici necessari e delle difficoltà connesse al loro trattamento (specie all'epoca delle missioni Apollo in cui Margaret e colleghi si trovavano a operare). Una trattazione esaustiva della matematica coinvolta nella progettazione di un volo interplanetario esula dagli obiettivi di questo articolo (e purtroppo dalle competenze delle autrici), ma ci sembra comunque opportuno, tramite la dicussione di qualche tema specifico, dare almeno il "sapore matematico" delle problematiche relative.

In generale, l'ambizione umana all'esplorazione dello spazio presenta difficoltà legate ai viaggi planetari, alle orbite di trasferimento e alla navigazione spaziale, che hanno portato una disciplina relativamente classica come la Meccanica celeste (tra i cui padri fondatori menzioniamo nomi illustri quali Newton, Eulero, Lagrange, Laplace, Poincaré – senza dimenticare Keplero come precursore) a sviluppare una nuova branca detta Astrodinamica, che ha come oggetto proprio l'orbita e il moto di corpi artificiali quali astronavi e satelliti nello spazio. Non è difficile, anche ragionando a spanne, immaginare quali siano le questioni implicate [9]: sebbene le forze gravitazionali generate dal Sole e dagli altri pianeti siano calcolabili, i viaggi spaziali sono caratterizzati da una combinazione difficilmente gestibile di alta velocità e limitato carburante, il che rende proibitivo realizzare variazioni significative della direzione e della velocità, che devono quindi essere previste con grande accuratezza fin dal momento del lancio. Gli ingredienti da utilizzare per programmare il volo di un velivolo spaziale sono essenzialmente due: la forza di gravità esercitata dal pianeta di partenza e da quello di arrivo, e la spinta generata dall'accensione di un propulsore. L'astronave deve incontrare il pianeta di arrivo al momento giusto e nel posto giusto: un Apollo deve arrivare nei pressi della Luna con la giusta velocità, altitudine e direzione. Oltre quindi a dover curare con estrema precisione le condizioni iniziali del decollo, è necessario monitorare in ogni istante tramite misurazioni ottiche o radar la traiettoria di volo o da Terra o dall'astronave stessa, con riferimento alla direzione e alla posizione delle stelle, facendo se necessario piccole ma continue correzioni. Il velivolo deve rallentare quando si avvicina alla Luna, acquisire di nuovo velocità per entrare nella sua orbita e decelerare infine durante l'atterraggio – pardon, l'allunaggio. Se si tratta di un volo con equipaggio umano, sarebbe opportuno che fosse poi in grado di ripartire e rientrare sulla Terra, il che per certi versi – in maniera forse controintuitiva – è la parte più critica del viaggio spaziale: se l'inclinazione di rientro è troppo ripida, la decelerazione eccessiva può far incendiare il velivolo; se invece è troppo stretta, il rischio è che l'astronave scappi fuori dall'atmosfera.

Riassumendo, diverse belle gatte da pelare. Non stupisce quindi né che la meccanica celeste abbia visto i contributi dei maggiori matematici applicati di tutti i tempi, né che il volo umano diretto alla Luna si sia sviluppato con grande difficoltà e in un momento in cui era strategico soprattutto per motivi politici.

Nel seguito, ci concentreremo su due dei problemi sopra menzionati, connessi tra loro per diversi motivi e strettamente legati alla nostra storia perché certamente all'epoca dell'Apollo erano oggetto di studio – e di preoccupazione – per chi lavorava alla pianificazione dei lanci: il problema degli incontri ravvicinati e quello delle patched conics.

La prima questione si pone tutte le volte che la nostra astronave si avvicina al corpo su cui vuole atterrare (o allunare). L'altra è preliminare all'allunaggio (o atterraggio), perché è la questione che regola le orbite in generale e quindi il passaggio dell'astronave dall'influenza della Terra a quella della Luna (e viceversa).

Il problema dell'appianetaggio (consentiteci un neologismo) è in sostanza quello della soluzione in tempo reale di un sistema differenziale "stiff" (letteralmente "rigido"), ovvero tale che i metodi numerici per risolverlo sono numericamente instabili, a meno che non si scelga un passo d'integrazione estremamente piccolo. Quando si integra numericamente un'equazione differenziale, ci si aspetterebbe che la dimensione del passo d'integrazione richiesto sia relativamente piccola nelle regioni in cui la curva della soluzione mostra molte variazioni ma possa essere grande quando la curva della soluzione è più liscia: nei problemi "stiff" invece, affinché un metodo numerico fornisca una soluzione affidabile è necessario che le dimensioni del passo siano inaccettabilmente piccole anche in una regione in cui la curva della soluzione è molto liscia.

Un problema di questo tipo, legato alla balistica esterna, aveva portato vent'anni prima di Apollo allo sviluppo di ENIAC, il primo computer elettronico, digitale, general purpose costruito nell'ambito di un progetto segreto del ministero della difesa degli Stati Uniti durante la seconda guerra mondiale [3]: al tempo di Margaret la capacità di calcolo disponibile a bordo (ma anche a Terra) rimaneva comunque troppo limitata, e quindi il conseguente problema di controllo poneva difficoltà enormi. Dai rapporti della NASA di quel periodo (ora disponibili al pubblico) è evidente un crescente interesse per la questione della regolarizzazione delle doppie collisioni nel problema degli N-corpi.

Vediamola nella sua formulazione più elementare: consideriamo due corpi, la Luna (o la Terra) e l'astronave, modellizzati da particelle di masse rispettivamente M e m. La mutua attrazione tra le due masse è governata dalla legge di Newton ed è data, in modulo, da

$$G\frac{Mm}{r^2}, \tag{0}$$

dove G è la costante di gravitazione universale e r è la distanza tra le due masse. Di conseguenza, la forza che la massa M esercita sulla massa m si può esprimere come

$$G\frac{Mm}{r^3}(y_M - y_m), \tag{1}$$

dove y_M e y_m sono le posizioni delle due masse in un sistema di riferimento inerziale. Per semplificare le cose, consideriamo un sistema di coordinate x_1, x_2, x_3 centrato nella massa M. Allora l'equazione differenziale che governa il moto della particella m rispetto alla massa centrale M è la seguente:

$$\ddot{x} + \frac{G(M+m)}{r^3}(x = 0), \tag{2}$$

dove $x = y_M - y_m$ è il vettore posizione della particella m nel sistema di riferimento centrato nella massa M.

Sulla particella m possono agire anche altre forze, oltre all'attrazione gravitazionale principale: può trattarsi per esempio di forze prodotte dal drag atmosferico, dalla pressione di radiazione solare, dall'attrazione da parte di un terzo corpo o dall'asfericità della massa M. Indicata con P la risultante di tutte queste forze, che chiameremo *forza di perturbazione*, l'equazione modificata del moto sarà quindi

$$\ddot{x} + \frac{K^2}{r^3}x = P, \text{ dove } K^2 = G(M+m).$$

Naturalmente, l'attrazione gravitazionale tra le due masse è infinita se $r = 0$, cioè se si verifica una collisione tra i due corpi. In altre parole, l'equazione (2) e la sua modificata sono singolari nell'origine, poiché l'attrazione gravitazionale newtoniana della massa centrale è infinita in quel punto. Sfortunatamente, questo fatto causa non solo difficoltà teoriche ma anche sgradevoli problemi pratici: se la particella si avvicina molto alla massa centrale, causando quella che viene chiamata una *quasi collisione*, vengono prodotte grandi forze gravitazionali e quindi variazioni importanti nell'orbita. Durante un'integrazione numerica l'unico modo per superare una difficoltà simile è analizzare la fase di avvicinamento tra le due particelle tramite intervalli piccoli e molti step di integrazione. Purtroppo, però, a causa degli errori di arrotondamento, la precisione numerica dopo una quasi collisione è inevitabilmente molto scarsa, come tipico in tutti i problemi stiff. Questo non è un grosso problema nelle questioni della meccanica celeste classica, dal momento che le collisioni dei corpi planetari sono piuttosto rare (anche se non impossibili: sulla Terra, per esempio, meteoriti grandi come quello che si ritiene abbia causato l'estinzione dei dinosauri cadono molto raramente, in ragione si stima di 1 ogni 100 milioni di anni, ma le collisioni di oggetti più piccoli che comunque possono fare qualche danno, come il meteorite di Chelyabinsk del 2013, sono molto più frequenti); al contrario, qualsiasi missione di un velivolo spaziale che parte da Terra e vuole arrivare a sbarcare su un altro pianeta implica un approccio ravvicinato sia all'inizio della missione che nell'approssimarsi alla destinazione. Dal punto di vista matematico, quindi, risolvere il problema sia del decollo che dell'atterraggio richiede necessariamente la trasformazione di equazioni differenziali singolari in equazioni regolari, in un processo che chiamiamo di *regolarizzazione* [si veda per esempio 13]. L'idea è cioè trasformare il campo vettoriale singolare in uno regolare in modo tale che quando due corpi passano abbastanza vicini (anche se non c'è collisione) i calcoli numerici siano più stabili. Parliamo di "regolarizzazione di Levi-Civita (1920)" quando lavoriamo nel piano, e di "regolarizzazione di Kustaanheimo-Stiefel (1964)" se consideriamo invece la situazione tridimensionale.

Senza entrare nei dettagli tecnici (per cui si rimanda a [8, 10, 16]), precisiamo che le trasformazioni usate in queste regolarizzazioni sono del tipo

$$x = u^2 \tag{3}$$

ma, mentre la trasformazione di Levi Civita in una dimensione usa variabili reali, quella nel piano usa variabili complesse, che quindi fanno riferimento a uno spazio parametrico in due dimensioni, mentre quella di Kunstaanheimo-Stiefel richiede due dimensioni ulteriori, ovvero richiede R^4 come spazio dei parametri: è in questo contesto che i quaternioni fanno ingresso in modo molto naturale. In estrema sintesi, comunque, la procedura di regolarizzazione si sviluppa in tutti e tre i casi (moto in una, due e tre dimensioni) con la stessa struttura, ovvero seguendo tre passaggi: in primis si applicano le trasformazioni (3) passando dalle variabili fisiche a quelle parametriche; poi si introduce un *tempo fittizio* che permette di vedere le equazioni del moto come rallentate (ovvero impedisce che la velocità diventi infinita nella singolarità); infine, per concludere, si usa la conservazione dell'energia in modo da trasformare le equazioni differenziali singolari in equazioni regolari, attraverso l'introduzione dello *spazio delle fasi esteso*. In pratica, all'ultimo passo si ottiene un'equazione lineare, l'equazione dell'oscillatore armonico in una, in due o in quattro dimensioni.

Risolto (si fa per dire) il problema dell'appianetaggio, concentriamoci su quello del trasferimento tra un pianeta e l'altro, niente affatto banale. Non lo sarebbe neppure se la nostra idea fosse semplicemente mandare un razzo a sbattere sulla superficie della Luna (come immaginava Jules Verne nella prima parte di *Dalla Terra alla Luna*); figuriamoci come aumentano le difficoltà nel caso di un volo con equipaggio, in cui non solo si spera di non andare a sbattere, ma si vorrebbe anche tornare indietro.

Cerchiamo anche stavolta di partire dall'inizio, scusandoci con chi fosse esperto di questi temi per l'enunciazione di concetti forse troppo elementari. Consideriamo il problema dei due corpi, in cui una massa M e una più piccola m sono attratte reciprocamente secondo la legge di gravitazione universale prima ricordata. Le soluzioni dell'equazione di Newton sono coniche: le orbite cioè della massa

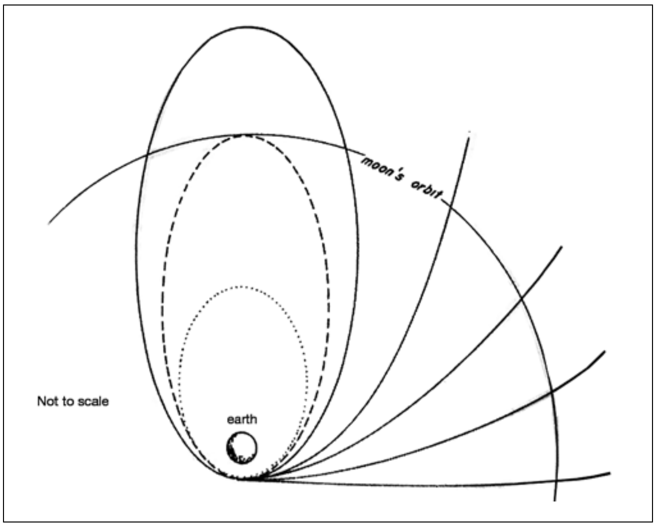

FIGURA 2 – Effetto della velocità di iniezione sulla traiettoria dell'orbita.

piccola sono circonferenze, ellissi, parabole o iperboli di cui la massa grande occupa un fuoco. Nel nostro caso, la massa grande è la Terra, mentre la massa piccola è l'astronave: se l'astronave non ha propulsione, naturalmente non si stacca dal suolo; immaginiamo quindi che un razzo propulsore l'abbia portata a una certa distanza dalla superficie della Terra, per poi lasciarla al suo volo libero. A questo punto, perché l'astronave non ricada sulla Terra bisogna che si muova con una velocità tale che la forza centrifuga compensi l'attrazione gravitazionale: il razzo, cioè, non solo deve sollevarla, ma la deve anche abbandonare con una velocità sufficiente a consentirle di restare in orbita. La forma dell'orbita dipende dalla posizione e dalla velocità dell'astronave all'inizio del suo volo libero, come possiamo vedere nella Figura 2.

Nel caso limite in cui la velocità di iniezione è infinita la traiettoria sarà una retta; a mano a mano che diminuiamo la velocità l'orbita passa da iperbolica a parabolica a ellittica. In particolare, considerando fisso il punto e la direzione di lancio (che è comunque tangente all'orbita nel punto di inserzione), tra due orbite ellittiche con uguale perigeo (il punto di lancio) ha un apogeo più distante quella che corrisponde a una maggiore velocità nel perigeo.

Supponiamo adesso di voler far "scappare" l'astronave dal campo gravitazionale terrestre, per esempio per farla viaggiare verso Marte: se continuasse a muoversi su orbite ellittiche con fuoco la Terra, non sfuggirebbe mai alla sua attrazione; di conseguenza, per realizzare la "fuga" dovremo darle abbastanza velocità da immetterla su un'orbita parabolica o iperbolica (nel sistema di riferimento Terra-centrico) con fuoco nel centro della Terra (*orbita di fuga*). Una volta "scappata" dalla Terra, si considera che l'astronave si muova solo influenzata dal campo del Sole, quindi per trasferirsi dall'orbita della Terra a quella di Marte serve un'orbita ellittica (nel sistema di riferimento eliocentrico) – che abbia un fuoco nel Sole e, come minimo, perielio sull'orbita della Terra e afelio sull'orbita di Marte. Perché l'orbita solare sia ellittica e non iperbolica, la velocita di iniezione dovrà essere superiore alla velocità di fuga dalla Terra, ma non dal Sole. All'arrivo nel campo gravitazionale di Marte, la nave si muove su un'orbita iperbolica (nel sistema di riferimento Marte-centrico) con un fuoco nel centro di Marte (che in questo caso si chiama *orbita di cattura*). Perché però ci sia effettivamente la cattura, al momento giusto ci si deve mettere in un'orbita ellittica attorno a Marte: ragionando all'inverso di quanto visto nel caso della fuga, si deve quindi "frenare" per passare dall'iniziale orbita iperbolica a una circolare o ellittica, sempre centrata su Marte – aspetto, quello dell'aerobraking, fondamentale nella pianificazione di una missione Terra-Marte. Nella Figura 3 rappresentiamo schematicamente la situazione.

In altri termini, il trasferimento della nostra astronave dalla Terra a Marte si ottiene facendo una sorta di collage di orbite coniche semplici con fuochi però in corpi celesti diversi: la tecnica rappresentata da questo collage è quella che tecnicamente si chiama delle *patched conics*.

La tecnica delle patched conics si può parafrasare in termini di "sfere di influenza" dei corpi celesti coinvolti. Definiamo *sfera di influenza* di un certo corpo celeste C come la sfera, centrata in C, all'interno della quale l'influenza gravitazionale primaria su un oggetto in orbita è C stesso. Ricordando che l'attrazione gravitazionale esercitata da un corpo su un altro è direttamente proporzionale alla sua massa e inversamente proporzionale al quadrato della distanza, è chiaro che finché l'oggetto in orbita si trova "vicino" al corpo C, la sua attrazione gravitazionale sarà prevalente rispetto a

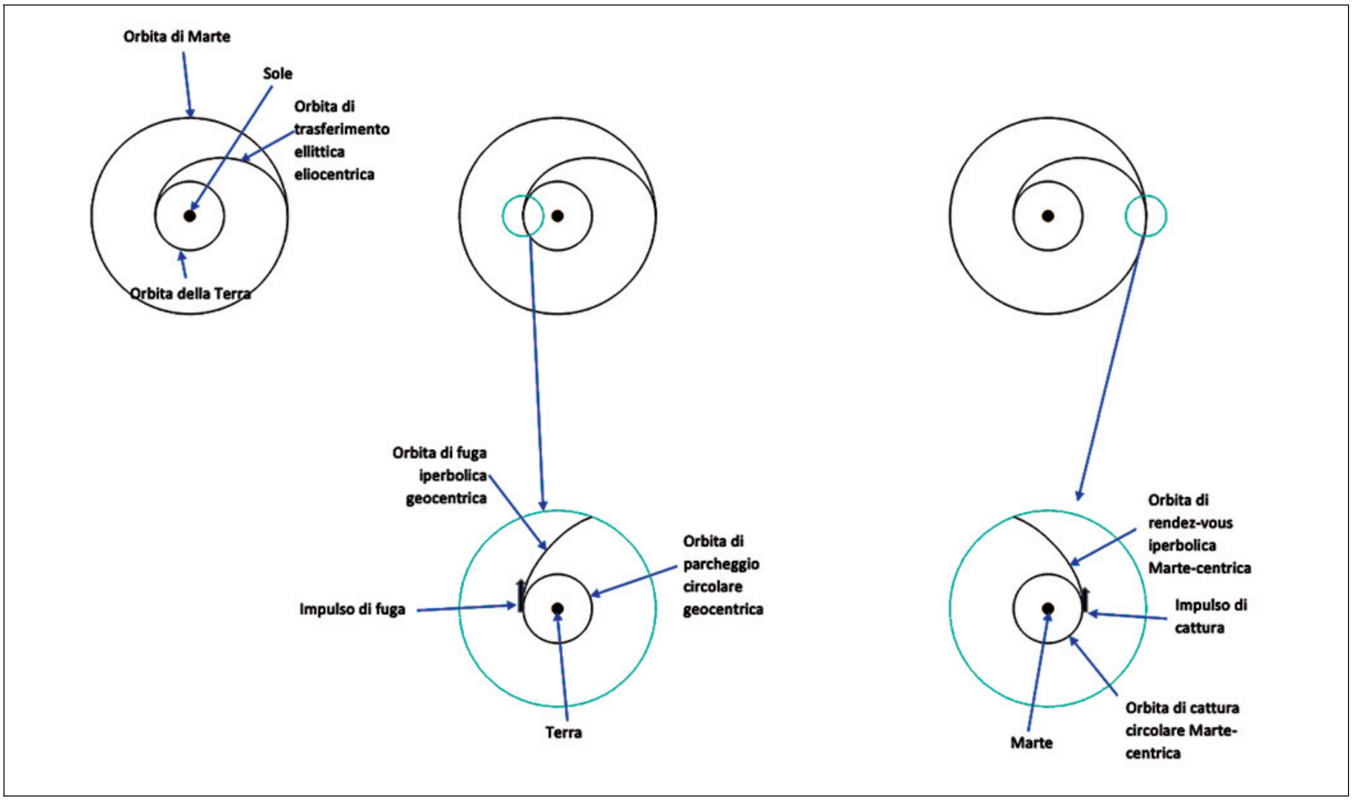

FIGURA 3 – Sistema di patched conics per il trasferimento Terra-Marte: la figura di sinistra rappresenta globalmente il processo di trasferimento, quella centrale il dettaglio alla fuga, quella di destra il dettaglio alla cattura.

quella di altri corpi celesti magari più massicci, ma molto più lontani. Consideriamo quindi un altro corpo celeste, C'. La presenza dei due corpi determina il raggio delle rispettive sfere di influenza: il raggio della sfera di influenza di C sarà pari alla distanza alla quale il valore assoluto dell'attrazione gravitazionale di C è uguale al valore assoluto di quella di C'. A questo punto il concetto di patched conics può essere interpretato come una semplificazione del problema degli n-corpi, nella quale si approssima la soluzione raccordando tra di loro varie soluzioni del problema più semplice dei 2 corpi: il punto di raccordo tra tratti conici successivi avviene in corrispondenza del confine delle rispettive sfere di influenza dei corpi.

Nel nostro esempio, il primo corpo celeste è la Terra e l'oggetto in orbita è il velivolo spaziale. In linea di principio, il velivolo è soggetto all'attrazione gravitazionale della Terra, del Sole, di Marte e di altri pianeti. Tuttavia, visto che l'attrazione esercitata da un corpo è direttamente proporzionale alla sua massa e inversamente proporzionale alla distanza, quando il velivolo si trova vicino alla Terra possiamo senz'altro trascurare l'influenza degli altri pianeti (che sono piccoli, oltre che lontani), e scrivere l'equazione del moto, nel sistema di riferimento geocentrico, in questi termini:

$$a = \text{(accelerazione centrale dovuta alla Terra)} +$$

$$+ \text{(accelerazione dovuta al Sole)} \sim \frac{M_T}{x^2} - \frac{M_S}{(d_{TS-x})^2},$$

dove M_T è la massa della Terra, M_S è la massa del sole e d_{TS} è la distanza Terra-Sole.

Finché x è piccolo, il secondo termine è trascurabile. A mano a mano che x cresce, il primo termine decresce rapidamente, mentre il secondo cresce rapidamente, fino a che non diventano dello stesso ordine di grandezza. È a questo punto che la sfera d'influenza della Terra lascia posto a quella del Sole, che governa il moto del velivolo fino a che, procedendo con un ragionamento inverso a quello appena visto, l'influenza del pianeta cui siamo diretti, in questo caso Marte, non diventa progressivamente dominante. La tecnica delle patched conics per una traiettoria Terra-Marte consiste quindi nel collage di due successive soluzioni del problema dei 2 corpi, una in corrispondenza del confine tra la sfera d'influenza della Terra e di quella del Sole, l'altra al confine tra la sfera d'influenza del Sole e quella di Marte.

Nel caso di un trasferimento Terra-Luna il collage è più semplice, perché non c'è bisogno di fuga dalla Terra – molto semplicemente perché la Luna è a sua volta in orbita attorno al nostro pianeta, quindi il trasferimento avviene all'interno del pozzo gravitazionale della Terra. Per arrivare quindi alla Luna basterà posizionarsi in un'orbita ellittica Terra-centrica (o meglio Terra-focale) con apogeo abbastanza grande da arrivare a intersecare l'orbita della Luna (vedi di nuovo la Figura 2): la velocità con cui il razzo che la "spinge" da Terra deve lasciare l'astronave al perigeo deve quindi essere tale da far sì che l'apogeo sia almeno pari alla distanza Terra-Luna.

Se mirassimo a un trasferimento "alla Verne", cioè ci accontentassimo di sbattere sulla superficie della Luna, non ci sarebbe bisogno di nessun collage di coniche: scelta la velocità iniziale tale che l'apogeo dell'orbita corrispondente sia maggiore della distanza Terra-Luna, basterebbe (per modo di dire) calcolare i tempi in modo tale da arrivare a intersecare l'orbita lunare esattamente nel momento in cui la Luna passa di lì. Questo non è facile, perché il sistema Terra-Luna presenta diverse complicazioni intrinseche, ma diciamo che con qualche cautela, relativa soprattutto a un'accurata scelta dell'angolo di partenza della traiettoria, ci si riesce.

Il collage è invece indispensabile in caso si voglia allunare con calma e senza distruggere l'astronave nell'impresa: in questo caso dovremo prima immetterci in una traiettoria ellittica centrata nella Terra con apogeo all'altezza della Luna, che corrisponda a una traiettoria iperbolica se vista nel sistema Luna-centrico, e poi "frenare" per metterci in un'orbita ellittica attorno alla Luna, quella del modulo di rientro da cui il buon Collins guardava i suoi colleghi Armstrong e Aldrin che erano scesi col LEM sulla sua superficie. Nella realtà, il problema del collage è complicato dalla presenza contemporanea di tutti i corpi celesti (in particolare la presenza della Terra non è affatto trascurabile mentre si orbita attorno alla Luna), ma in questo modo si ottiene una prima approssimazione che si può calcolare a mano.

Nel caso dell'Apollo 11 in realtà il patching è stato un po' più complicato, e questo essenzialmente per ragioni economiche che hanno richiesto l'implementazione di adeguate soluzioni tecniche. Immaginiamo infatti di voler mettere in pratica lo schema del volo diretto (magari con qualche cautela legata a come impatteremo sulla superficie lunare): in linea di principio, per farlo avremmo bisogno di una sola sonda composta da un motore che rimanga con noi tutto il viaggio, in modo da consentirci di allunare e da poterci poi spingere adeguatamente per acquisire velocità sufficiente di ritorno. Fatti due conti, una sonda siffatta raggiungerebbe il ragguardevole peso di 74 tonnellate, un carico troppo pesante per il razzo che dovrebbe metterla in orbita all'andata – "andare in orbita costa un tanto al chilo" pare sia il monito con cui vengono formati i buoni ingegneri aerospaziali, abituati per questo ad aspirare alla leggerezza di tutto quello che vogliono far orbitare. Visto che una sola sonda è troppo pesante, il piano B potrebbe essere lanciarne due in contemporanea, una composta dal modulo per allunare e una da quello per tornare indietro: una volta messi i due pezzi in orbita terrestre, bisognerebbe ricombinarli con una manovra d'incontro (detta tanto tecnicamente quanto romanticamente *rendezvous*) per poi farli dirigere insieme alla Luna e tornare indietro insieme a fine missione. Purtroppo il piano B, raddoppiando il numero di razzi necessari, risulterebbe comunque troppo dispendioso. La soluzione giusta, per quanto un po' macchinosa, fu infine individuata da John Humbolt, ingegnere della NASA fino a quel momento sconosciuto ma poi passato alla storia come l'inventore del *parcheggio lunare*. I tre componenti dell'equipaggio dovevano partire nel modulo di comando, attaccato al modulo di servizio, e agganciare opportunamente il modulo lunare (il summenzionato LEM, acronimo di Lunar Excursion Module) che alla partenza, nell'ogiva del razzo, era sistemato sotto il modulo di servizio. Insieme, dovevano arrivare alla Luna dove il LEM si sarebbe separato per atterrare, con a bordo Armstrong e Aldrin, mentre il modulo di comando, attaccato a quello di servizio, sarebbe rimasto alla guida di Collins ad aspettare in orbita di parcheggio lunare. Una volta effettuato il *rendez-vous* di ritorno tra il LEM e il resto dell'astronave, e rientrati Aldrin e Armstrong a bordo del modulo di comando, il LEM sarebbe stato abbandonato e gli altri due moduli avrebbero intrapreso il viaggio di ritorno verso l'atmosfera terrestre, nei pressi della quale anche il modulo di servizio sarebbe stato abbandonato per lasciar ammarare solo quello di comando contenente i tre eroi.

Dopo l'ultima visita sovietica del 1976, la Luna passa di moda per circa quindici anni: l'attenzione delle agenzie spaziali si rivolge ad altri corpi del sistema solare (Viking 1 e 2 indagano Marte, Voyager 1 e 2 visitano Giove, Saturno, Urano e Nettuno per poi puntare, a quasi quarant'anni di distanza, «verso l'infinito e oltre»). È solo il nuovo millennio a portare una rinascita dell'interesse per la Luna, caratterizzata però da profonde differenze col clima che aveva avvolto le missioni dell'epoca della Guerra fredda: puntano la Luna altri paesi (Europa, Giappone, India, Cina); pianificano missioni non solo le agenzie governative ma anche i privati (vedi Elon Musk); ma, soprattutto, si guarda alla Luna non come un approdo dove fare toccata e fuga, bensì come a un satellite da colonizzare e in cui installare avamposti umani più o meno permanenti, (forse) come prova generale per una fantascientifica migrazione verso altri pianeti. Questo pone ulteriori questioni fisiche, matematiche e ingegneristiche di enorme interesse: la superficie della Luna è decisamente un postaccio dove far risiedere una colonia di esseri umani – pur essendo scienziati. Per avere qualche chance, bisogna proteggere questi moderni pionieri da radiazioni, micro-meteoriti, lunamoti e ambiente termico particolarmente inospitale.

Come se non bastasse, l'idea naïf di portare da terra i materiali di costruzione è del tutto impraticabile (ricordate il motto del perfetto ingegnere aerospaziale?): per questo la ricerca in questo campo si sta concentrando sull'utilizzo delle risorse in situ, per ridurre al minimo la quantità di materiale che deve essere portato dalla Terra e quindi rendere la missione economicamente più sostenibile.

Ma questa è decisamente un'altra storia (raccontata dettagliatamente in [2]). Torniamo quindi a Margaret e a quanto abbiamo appreso dall'esperienza di Apollo.

4. – Non solo Apollo: gli albori della programmazione

Nelle testimonianze di coloro che scrissero e fecero funzionare i primi programmi per computer si riscontrano molte esperienze simili e abbastanza curiose, a partire dai metodi con cui avveniva la selezione per l'assunzione. Era difficile immaginare quali competenze servissero e così chi doveva scegliere si dimostrò alquanto creativo. Il primo computer *general purpose* funzionante fu l'ENIAC, progettato e programmato durante la seconda guerra mondiale. Per programmarlo furono selezionate, dal gruppo di giovani matematiche che lavoravano manualmente al calcolo delle tavole balistiche per conto del ministero della difesa, cinque ragazze, individuate come le più brillanti: per sceglierle venne loro richiesto anche se avevano paura della corrente elettrica! Certo quell'enorme computer pieno di luci intermittenti, spinotti e cavi da trasportare e connettere poteva incutere paura, ma la relazione tra questa domanda e la reale inclinazione a programmare forse non è proprio così evidente [15, 3].

Successivamente, quando furono definiti i primi linguaggi di programmazione, la conoscenza di uno di questi era sufficiente a garantire l'assunzione. Il che è grossomodo come dire che se sai il cinese puoi scrivere un romanzo in quella lingua. Questo perché chi avrebbe dovuto selezionare l'inclinazione a programmare non aveva la minima idea di quali fossero i requisiti necessari, né di come verificarli [5, 14].

Gli ingegneri dell'*hardware*, quelli che i computer li progettavano e li costruivano fisicamente, non erano interessati alle problematiche della programmazione che per loro restavano un mistero e volentieri delegavano ad altri tutti i problemi relativi al *software*, cioè i programmi che quel computer dovevano far funzionare. Per questo motivo vennero assunte tante donne e tanti giovani e talvolta, a causa di test automatici, anche donne afro-americane: il punteggio ottenuto al test superava in un balzo tutti i pregiudizi di razza o di genere. Tanto il lavoro di programmazione era d'importanza secondaria, solo un lavoro di supporto, adatto alle donne, considerate alla stregua di segretarie super specializzate!

Un altro fattore importante fu la completa libertà concessa ai giovani programmatori: tutti lavoravano partendo quasi da zero, provando e sperimentando nuove tecniche, imparando dagli errori, basandosi sul passa parola, ma senza far ricorso a regole imposte o prestabilite. Quello che veniva loro richiesto era soltanto impegno e dedizione, e soprattutto che i programmi funzionassero. Questa situazione, unita al fatto che l'attività di programmare è stimolante e divertente e che i salari erano buoni, fu il motore di gruppi di programmatori molto motivati e entusiasti del loro lavoro, che riuscirono magistralmente in varie imprese degne di nota. Al tempo non ebbero tutto il riconoscimento dovuto, soltanto di recente alcune storie sono state riscoperte e valutate nel modo giusto.

Gli ingegneri del software cominciarono ben presto a richiedere l'attenzione di quelli dell'hardware perché modifiche anche piccole dell'hardware avrebbero potuto portare miglioramenti sostanziali nel software. La voce dei programmatori cominciò a farsi sentire sempre più e a entrare a pieno diritto anche nella progettazione dell'hardware.

Bisogna aspettare la fine degli anni '60 perché la programmazione cominci a ricevere la giusta attenzione. A questo contribuì un libro meraviglioso composto di vari volumi (ancora in fase di completamento) scritto da Donald Knuth, uno dei padri dell'informatica mondiale, che s'intitola: "The Art of Computer Programming"[11]. La programmazione del computer richiede una competenza molto particolare, l'importanza e la diffusione dei computer continuano a crescere e non è più sufficiente che i programmi funzionino, devono essere ben scritti e documentati e soprattutto essere efficienti. Insomma occorrono estro, intuizione e originalità, oltre alle competenze tecniche. Programmare diventa un'arte.

5. – Mettere a frutto l'esperienza

In un recente discorso tenuto come conferenza su invito a un importante convegno di software engineering, Margaret ha tirato le fila della sua incredibile avventura nella programmazione che si è protratta per circa 60 anni, sottolineando gli aspetti che nel tempo sono cambiati e quelli che si sono mantenuti, nonché le problematiche fondamentali [18].

Durante la sua lunghissima esperienza ha avuto modo di confrontarsi, parole sue, con ogni specie di errore possibile di programmazione; ha ben presto capito che il problema maggiore di un ingegnere del software è quello di produrre programmi affidabili, compito tanto più arduo quanto più i sistemi sono complessi. Quando ci sono più di cento persone che lavorano a un progetto il problema del coordinamento e delle responsabilità, soprattutto per gli aggiornamenti che possono cancellare dati utili, è fondamentale. Con uno spiccato senso dell'umorismo Margaret ci racconta come ha dovuto imparare a prevenire gli errori: quando un tuo errore fa bloccare tutto il sistema e il lavoro di tantissime persone, che inferocite ti vengono a urlare, impari velocemente a testare il tuo programma in locale, prima di inserirlo nel programma complessivo! La documentazione è importantissima, va fatta giorno per giorno. «Sempre aspettarsi l'inaspettato» raccomanda, l'errore arriva nei modi più impensati. Nel suo gruppo avevano escogitato varie tecniche per evitare gli errori: una era quella di classificarli associando a ogni singolo tipo di errore la foto di chi l'aveva commesso per primo, cosa che doveva senz'altro restare impressa. Margaret aveva una grande responsabilità e una vera ossessione per gli errori, che la teneva sveglia la notte, ma che l'ha condotta per altro, a prevedere un sistema affidabile, in grado anche di fare alcune correzioni, come quello dell'Apollo 11.

Negli anni successivi alle missioni Apollo, Margaret decise di mettere a frutto la sua incredibile esperienza di programmatrice, lasciò l'MIT e la NASA e creò la sua compagnia, la Hamilton Technologies. L'esperienza vissuta con le missioni lunari le aveva insegnato tante cose e soprattutto che i sistemi in natura non sono governati da un singolo agente, ma da una moltitudine di agenti, che agiscono contemporaneamente in maniera non coordinata e reagiscono in base a eventi specifici che possono verificarsi nel sistema. I linguaggi che avevano usato erano del tutto inadeguati perché erano definiti per sistemi centralizzati e sequenziali. Un linguaggio di programmazione avrebbe dovuto dare la possibilità di definire queste caratteristiche e mettere a disposizione dei meccanismi per realizzarle, per evitare così di dover esplicitamente organizzare la sequenza di istruzioni per ogni possibile evento che si presentasse. Inoltre aveva compreso che ogni sistema è inerentemente un sistema di sistemi.

La Hamilton Technologies studiò e mise a punto un linguaggio per la definizione di sistemi, The Universal System Language *(USL)*, basato sulla teoria del controllo dei sistemi. A differenza dei linguaggi tradizionali si basa sulla filosofia della prevenzione: invece di individuare nuovi modi di testare l'errore, cosa che avviene troppo avanti durante il ciclo di vita del programma, impedisce gli errori subito all'inizio, al momento della definizione del sistema. Il linguaggio non è un vero e proprio di linguaggio di programmazione, piuttosto un linguaggio formale sul tipo di quelli strettamente teorici.

Il linguaggio USL è basato su una serie di assiomi e di regole formali. Lo stesso linguaggio è usato per definire cose diverse come:

- Tutti gli aspetti di un sistema e la sua evoluzione.
- Tutte le architetture funzionali, le risorse e le allocazioni, l'hardware, il software e i prodotti umani compresi.
- La documentazione.
- Tutte le definizioni.

Il linguaggio è indipendente dalla sintassi, dall'implementazione e dall'architettura. Pur essendo formale a differenza di altri linguaggi formali più matematici è *user-friendly*.

Quando un sistema è definito con un linguaggio di questo tipo, si può subito analizzare automaticamente e verificare che sia stato formalizzato correttamente; a questo punto anche la maggior parte del progetto e del codice stesso può essere ugualmente generata in modo automatico. Così il modello ottenuto può essere mandato in esecuzione e, se non soddisfa i requisiti voluti, si può rivedere la definizione fin quando non sarà perfetto. La filosofia del linguaggio è di costruire sistemi affidabili composti a loro volta di sotto sistemi affidabili. I sistemi riconosciuti affidabili possono essere riusati e formare la libreria del linguaggio.

In conclusione, se il linguaggio è usato correttamente la maggior parte degli errori viene evitata, inclusi quelli di interfaccia e quelli che ne derivano, in quanto l'analizzatore va a caccia di tutti gli errori causati da uno scorretto uso del linguaggio. Con la verifica automatica non c'è più bisogno di eseguire la maggior parte del *testing*.

Secondo Margaret, sono stati proprio gli errori a mostrare la strada per evitarli. Dunque un linguaggio adeguato per la definizione di un sistema complesso può servire esso stesso come ingegnere del software e con un paradigma di controllo preventivo degli errori conduce direttamente verso il futuro.

Per il suo contributo alla scienza in generale, alle missioni spaziali e allo sviluppo dei linguaggi di programmazione in particolare, Margaret Hamilton è stata premiata dal presidente degli Stati Uniti Barack Obama nel 2017 con la medaglia della libertà, portando agli onori della cronaca la storia dei suoi straordinari successi, fino ad allora non abbastanza riconosciuti.

BIBLIOGRAFIA

[1] Bate, R.R., Mueller, D.D., White, J.E.: Fundamentals of astrodynamics, Dover Publ. (1971)
[2] Benvenuti, S., Ceccanti, F., De Kestelier, X.: Living on the Moon: topological optimization of a 3D-printed lunar shelter, Nexus Network Journal vol. 15, issue 2, p. 285-302 (2013).
[3] Benvenuti, S., Pagli, L.: Refrigerator ladies, Matematica, Cultura e Società. Rivista dell'Unione Matematica Italiana, 1, 51-64 (2016).
[4] Benvenuti, S., Pagli, L.: From the Earth to the Moon: two stories of Women and Maths, to appear in Faces of Geometry: from Agnesi to Mirzakhani, Magnaghi-Delfino P., Mele G., Norando T. (a cura di), Springer (2020)
[5] Bodei, C., Pagli, L.: L'informatica non è un paese per donne Mondo Digitale (2017).
[6] Cameron, L.: What to Know About The Scientist Who Invented The Term: Software Engineering. IEEE Computer Society (2020).
[7] Caraveo, P.: Conquistati dalla Luna. Storia di un'attrazione senza tempo, Cortina (2019).
[8] Celletti A.: Basics of regularization theory, in: Steves B.A., Maciejewski A.J., Hendry M. (eds) Chaotic Worlds: From Order to Disorder in Gravitational N-Body Dynamical Systems. Springer (2006).
[9] Chau, K.T.: Application of differential equations in Engineering and Mechanics, CRC press (2018)
[10] Flandoli, I.: Teoria della regolarizzazione e Quaternioni, tesi di Laurea cortesia del relatore Giacomo Tommei (2014)
[11] Knuth, D.: The Art of Computer Programming, Addison Wesley, v.1, 2, 3, 4 (1968-2019).
[12] Light, J.S.: When Computers Were Women. Tecnology and Culture 40(3), RLC (1999).
[13] Stiefel, E.L., Scheifele, G.: Linear and regular celestial mechanics, Springer-Verlag (1971)

[14] THOMPSON, C.: The secret History of Women in Coding, New York Times, Febbraio 2019.
[15] Top Secret Rosies: The female "Computers" of WW II. documentario diretto da LeAnn Erickson, PBS Studio (2010).
[16] WALDVOGEL J.: Quaternions for regularizing Celestial Mechanics: the right way, Cel. Mech. Dyn. Astr. vol. 102, pp. 149162 (2008)
[17] https://en.wikipedia.org/wiki/Margaret_Hamilton_(software_engineer).
[18] M. HAMILTON ICSE 2018 – Plenary Sessions (https://www.youtube.com/watch?v=ZbVOF0Uk5lU).

Silvia Benvenuti

Silvia Benvenuti è professore di Matematiche complementari presso l'Università di Bologna. Dal 2006, quando ha conseguito il titolo di Master in Comunicazione della scienza della SISSA di Trieste, si occupa di comunicazione della matematica. È autrice di monografie e articoli di comunicazione scientifica per riviste quali Linx Magazine, *XLaTangente, Mate, Maddmaths!, Prisma. Partecipa da anni a trasmissioni divulgative di Rai 3 e Rai Scuola. È membro del* Centro matematita *e del comitato RPA (Raising Public Awareness) della European Mathematical Society – di cui è anche vicepresidente, e della commissione comunicazione dell'Unione Matematica Italiana.*

Linda Pagli

Linda Pagli, già Professore Ordinario di Informatica presso l'Università di Pisa, è in pensione dal gennaio 2021. Ha trascorso diversi periodi come visiting scientist presso la Carleton University e la Ottawa University in Canada. È stata ricercatrice presso l'Accademia Italiana della Columbia University di New York e professore in visita presso l'Università del Botswana. La sua attività di ricerca, iniziata nel campo degli algoritmi e delle strutture dati, è stata successivamente dedicata alle basi del calcolo in diversi modelli, come i modelli paralleli, distribuiti, e modelli con agenti mobili e robot.
Per conto dell'Unesco e del Ministero degli Affari Esteri italiano ha svolto un'intensa attività formativa per docenti di informatica nei paesi in via di sviluppo, insegnando materie teoriche di base e attività di laboratorio in Giordania, Siria, Tunisia, Nigeria, Cile, Yemen e Cuba.
È coautrice di quattro libri e diversi articoli dedicati alla divulgazione dell'informatica. Negli ultimi anni ha iniziato a considerare il problema della defezione delle studentesse in Informatica e ha pubblicato alcuni lavori preliminari su questo argomento.