

DEMETRIO RIA

*Ricercatore a tempo determinato di Pedagogia Sperimentale
Università del Salento*

Dal computational thinking al systemic learning

Riassunto

Le tecnologie dell'informazione e della comunicazione (TIC) sono portatrici di un cambiamento trasformatore della società. Per una società sostenibile è necessario, però, trasformare il modo di pensare oltre che quello di ordinare, conservare e interagire con l'informazione. I professionisti della *computer science* (CS) hanno già un kit di strumenti concettuali finalizzati alla soluzione dei loro problemi. Questo kit, a volte, viene definito genericamente come "pensiero computazionale". Il pensiero computazionale, però, tende a vedere il mondo in termini di una serie di problemi (o di problemi-tipo) che hanno soluzioni computazionali o comunque possono essere ricondotti a modelli risolutivi noti. In questo lavoro, si cerca di argomentare a sostegno di una idea di ponte necessario che unisca il pensiero computazionale alla pratica della sostenibilità. La sostenibilità, intesa in senso sistemico, è un approccio che consente di riconoscere la stretta interrelazione-interdipendenza tra tecnologia, comportamento umano e ambiente. Nelle pagine che seguono si cercherà di sviluppare, sia pure come abbozzo, una idea che fornisca una ontologia regionale pedagogica della relazione sostenibile tra comunità, ambiente e tecnologie ("mereologia" Putnam, Varzi Ferraris). In particolare si indagheranno alcune questioni sollevate dal computational thinking (CT) nell'ambito dello sviluppo delle intelligenze.

Abstract

The information and communication technologies (ICT) are the bearers of a transformational change in society. For a sustainable society is necessary, however, to transform the way we think as well as what to order, store and interact with information. The professionals of computer science (CS) already have a conceptual tool kit aimed at the solution of their problems. This kit, at times, is defined generically as "computational thinking." The computational thinking, however, tends to see the world in terms of a number of problems (or problems-type) that have computational solutions or however can be traced to known model solutions. In this paper, we try to argue in support of an idea of necessary bridge that unites computational thinking to the practice of sustainability. Sustainability, understood in a systemic way, it's an approach that allows you to recognize the close interdependence between technology, human behavior and the environment. In the following pages we will try to develop, even as a sketch, an idea that provides a

pedagogical regional ontology of the sustainable relationship between the community, the environment and technology ("mereology" Putnam, Varzi Ferraris). In particular we will investigate some of the issues raised by the computational thinking (CT) in the development of intelligence.

Parole chiave: Pensiero computazionale, Sistemi computazionali, Computer e società

Key words: Computational thinking, Systems Thinking, Computers and Society.

1. Introduzione

Nel corso degli ultimi decenni abbiamo assistito alla notevole crescita delle tecnologie dell'informazione e della comunicazione (TIC). È un dato, infatti, che tanti canali d'infra-strutture software hanno generato il loop di feedback del nostro comportamento sociale. Di fatto l'uso delle tecnologie non è stato sostenuto da un cambiamento trasformatore integrante e questo ha vanificato ogni possibile tentativo di adottare stili di vita più sostenibili.

Il CT potrebbe essere la via per superare questa fase deviante, anche se deve opportunamente arricchirsi di una visione/approccio sistemico. Da un certo punto di vista il CT porta con se importanti fattori soglia che in un'ottica pedagogica meritano adeguata attenzione. A un livello "macro", infatti, il CT potrebbe non consentirci di comprendere e di agire le trasformazioni necessarie per raggiungere la sostenibilità. A un livello "meso" frenerebbe la capacità di sviluppare adeguatamente la crescita equilibrata delle intelligenze, emotive e creative in primis, ma inficia anche quelle razionali e relazionali. Infine, a un livello "micro" si tenderebbe a ridurre il problem solving a una questione *sintattica*. Ciò

evidentemente allontana sia dalla dimensione del finding, posing e setting, sia dalla successiva funzione decisionale.

Partendo dalle idee chiave promosse dal CT e dalle osservazioni critiche degli stessi scienziati informatici, si cercherà di mostrare come sia possibile tracciare un confine più netto. Integrare l'approccio sistemico nella progettazione di percorsi formativi a sostegno della capacità di risolvere problemi. Soprattutto potrà essere utile avere davanti a se i confini territoriali di una metodologia estremamente ricca e al tempo stesso pervasiva cogliendone, tra l'altro, una particolare regionalità pedagogica.

2. Il pensiero computazionale

Cos'è il pensiero computazionale? Gli informatici tendono ad affrontare i problemi in un modo particolare. Cercano soluzioni algoritmiche in termini di manipolazione dei dati e di controllo di processo. Jeanette Wing, in un documento nel 2006, che ha fatto epoca, ha definito il pensiero computazionale sostenendo che questo può essere considerato il più importante contributo dell'informatica al mondo. Spingendosi fino a dichiarare che questo modo di pensare dovrebbe essere insegnato a tutti gli studenti di tutte le discipline. Wing ha fatto molti esempi su come gli scienziati informatici affrontano i problemi. Ha detto: "Quando tua figlia va a scuola la mattina, mette nel suo zaino le cose di cui ha bisogno per la giornata; questo è prefetching (recuperare) e caching. Quando tuo figlio perde i suoi guanti, e gli suggerisci di

tornare sui suoi passi; questo è back-tracking”. (Wing, 2006). Estrapolando da questi esempi, il messaggio generale è che gli scienziati del computer hanno una serie di strumenti e di metodi per generare la corrispondenza tra situazioni problematiche e tipi standard di soluzione. Inoltre, un aspetto altrettanto importante è che utilizzano una terminologia standard per descrivere questi modelli astratti di problem-solution.

A seguito di queste riflessioni e al dibattito (anche politico) che si è generato, la comunità informatica degli Stati Uniti ha prontamente adottato il pensiero computazionale come un importante obiettivo culturale. Per queste ragioni, ad esempio, la Computer Science Teachers Association (CSTA) ha istituito una task force per esplorare e diffondere l'insegnamento e l'apprendimento del pensiero computazionale. L'opinione considerata migliore, quella più diffusa e condivisa considera il pensiero computazionale come un processo di problem solving. Questi considerano che il CT sia (anche non si limita a) la capacità di formulare problemi in modo che un computer ci possa aiutare a risolverli. Ciò richiede che si sviluppino alcune capacità:

1. Organizzare e analizzare dati.
2. Rappresentare dati attraverso astrazioni, modelli o simulazioni.
3. Automatizzare attraverso il pensiero algoritmico le soluzioni.
4. Identificare, analizzare e combinare le misure e le risorse.

La CSTA aggiunge che queste abilità sono supportate da una serie di atteggiamenti più ampi. Tra cui: la capacità di affrontare la

complessità e problemi indeterminati, la tolleranza per l'ambiguità, l'abilità di lavorare con gli altri per raggiungere un obiettivo comune. Il pensiero computazionale si concentra sull'applicazione al problema di un insieme di astrazioni. A questa capacità di pensare con molti livelli di astrazione si è aggiunto l'utilizzo della matematica per lo sviluppo di algoritmi (Denning, 2009).

Tuttavia, dal momento in cui il concetto è stato introdotto, c'è stato ben poco dibattito critico su di esso. Le poche osservazioni che sono state scritte tendono a concentrarsi sulla vaghezza del termine (Jones, 2011), o sulla preoccupazione di non ridurre tutta la computer science ad uno solo dei suoi strumenti pratici. In questo senso, molto probabilmente, potremmo trovarci nella condizione come sostiene Maslow che “se tutto quello che hai è un martello, allora tutto sembra un chiodo” (Maslow, 1966). In altre parole, colui che opera con il pensiero computazionale cerca soltanto quei problemi che possono essere affrontati con i computer. Immediatamente questo fornisce una lente selettiva attraverso cui osservare il mondo. I problemi di cui è improbabile avere soluzioni computabili (ad esempio dilemmi etici, giudizi di valore, cambiamento sociale, ecc.) vengono ignorati.

Quello che emerge, in fondo, è che il CT si pone da un punto di vista intrinsecamente riduzionista. I problemi computazionali sono affrontati riducendoli a un insieme di variabili discrete che possono essere mappate su tipi di dati astratti, e una serie di fasi algoritmiche utili a manipolare questi dati. Sta di fatto che nel

processo, prospettive multiple sulla natura del problema si perdono, così come qualsiasi conoscenza contingente locale circa la “situazione” problematica (Gougen, 1992, Easterbrook 1993). Nella definizione del CSTA la complessità che viene considerata è quella algoritmica, che offre poco alla comprensione del più ampio studio di sistemi adattativi complessi (Holling, 2001). Manson suggerisce che “la complessità algoritmica offre due contributi relativamente accessori alla teoria complessità generale”(Manson, 2001). Da una parte misura lo sforzo necessario per risolvere un problema matematico, e la comprensione dei limiti di memorizzare e comunicare i dati forniti dalla teoria dell'informazione. Egli, però, identifica altre due forme di complessità necessarie per la comprensione del comportamento dei sistemi complessi, non lineari. Da una parte la complessità deterministica della teoria del caos e delle catastrofi. Dall'altra la complessità aggregata che sorge con il comportamento emergente dell'interazione di molti componenti all'interno di un sistema. Entrambe queste forme di complessità possono essere utilmente studiate attraverso i mezzi computazionali, principalmente attraverso l'utilizzo di modelli di simulazione. Tuttavia, il CT, almeno come definito in letteratura, non fornisce un opportuno insieme di concetti con cui è possibile realizzare tutto questo. La riduzione dei problemi ai loro componenti computazionali conduce spesso verso pratiche che minano la sostenibilità. Si consideri per esempio, l'importanza attribuita all'ottimizzazione

dei processi che sorge a causa del modo in cui si trattano le tecniche di scaling. Se si risolve un problema riducendolo a un insieme di astrazioni di base, è necessario scalare la soluzione computazionale di fronte a due risorse fundamentalmente limitate: spazio di memoria e di tempo del processore. Quindi, gli informatici sono addestrati per individuare soluzioni computazionali che si ottimizzino per spazio e tempo. La gestione del traffico in una città moderna offre un esempio illustrativo. Tramite il CT il problema della congestione del del traffico viene approcciato come un problema di ottimizzazione: la congestione può essere ridotto ottimizzando flusso di traffico. L'idea è che se vengono raccolti più dati circa i volumi di traffico, sarà possibile regolare dinamicamente i segnali stradali ad ogni incrocio. Purtroppo, questa soluzione di solito ha l'effetto opposto: aumenta la congestione.

Morozov prende di mira tutto questo nel suo recente libro *To Save Everything, Click Here: The Folly of Technological Solutionism* (Morozov, 2013). Egli definisce il “Technological Solutionism” (TS) come quella convinzione che problemi sociali complessi possono essere ri-espressi attraverso porzioni ben definite con soluzioni calcolabili oppure con processi trasparenti ed evidenti facilmente ottimizzabili. In un certo senso con i giusti algoritmi tutti i problemi possono essere “calcolati”. La riflessione di Morozov è più sottile di un semplice osservazione sui limiti del pensiero computazionale. L'idea chiave è che il nostro rapporto

con la tecnologia ri-modella la nostra “infrastruttura di problem-solving” in modi inaspettati. Per esempio: egli ritiene che l'idea di “gamification” (cambiare i comportamenti sociali, offrendo premi e incentivi, per trasformarlo in un gioco) sia particolarmente pericolosa. Sostiene che favorendo questo modello ci si allena a pensare in termini di “regolazione” del singolo cittadino. Oppure, l'idea che siamo in grado di ottimizzare i processi sociali di raccolta e analisi di sempre più grandi insiemi di dati sul comportamento umano è anch'essa pericolosa. Sempre Mozorov sostiene che ci si allena a credere che le soluzioni ai nostri problemi possono essere trovate attraverso un modello automatizzato di corrispondenza dei comportamenti. Perdendo di vista completamente che il comportamento si colloca all'interno di un processo cooperativo di co-costruzione della comprensione delle forze che lo modellano.

Inoltre, il dato è poi esaltato dal successo di questi approcci su piccoli e limitati problemi. Ciò tende a far aumentare la nostra fiducia nel TS, riducendo così la nostra capacità di affrontare le sfide più importanti. Il termine “solutionism” serve anche a sottolineare che pensare in termini di problemi e di soluzioni in sé è spesso controproducente. Rittel e Webber (1973) sostengono che molte questioni sociali sono meglio pensate come dilemmi, a cui dobbiamo rispondere in modo intelligente, piuttosto che come i problemi da risolvere. Quindi ci si colloca tra problemi “addomesticabili”, che hanno una chiara, definitiva formulazione

e possibilità di una soluzione oggettivamente corretta, e “dilemmi” che sono profondamente radicati in una situazione complessa questi ultimi rappresentano meglio la complessità del mondo nel quale viviamo. I dilemmi, infatti, godono delle seguenti proprietà:

1. non hanno una formulazione definitiva. Spesso, l'unico modo per comprendere appieno il problema è quello di tentare di risolverlo, e quindi non lo possiamo indicare chiaramente in anticipo.
2. non hanno una regola di arresto. Poiché non c'è fine alle catene causali che legano i sistemi aperti, l'unico limite alla soluzione di è un giudizio di valore: "questo è quanto si può ragionevolmente raggiungere".
3. non è possibile valutare completamente una soluzione in qualsiasi periodo di tempo ragionevole.
4. ogni soluzione è un'operazione one-shot. Ogni soluzione tentata cambia la natura del problema, quindi non c'è possibilità di tentare d'imparare per tentativi ed errori.
5. non è possibile elencare tutte le soluzioni possibili.
6. ogni problema è essenzialmente unico. Mentre un problema specifico potrebbe essere simile ai precedenti, i fattori contestuali locali variano, e questi sono abbastanza significativi per impedire soluzioni generiche da lavorare.
7. ogni dilemma può essere trattato come un sintomo di un altro problema.

8. ci sono diversi modi di spiegare la natura del problema. La scelta di come descrivere il problema determinerà la natura delle soluzioni accettabili. Inoltre, la scelta di spiegazione è solitamente determinata dalla visione della persona che descrive il problema.

Va precisato che il mancato studio dei dilemmi è diffuso in molte discipline¹.

Quindi si può ragionevolmente affermare che un eccessivo affidamento sul pensiero computazionale rischia di portare non solo a soluzioni computazionali che ignorano la sostenibilità sociale e ambientale, ma più spesso la minano attivamente.

3. Integrare il CT con il PENSIERO SISTEMICO

Il nucleo epistemologico della tesi qui sostenuta parte dal dato che il CT non ha completato la sua trasformazione in un vero e proprio sistema di pensiero. I sistemi di pensiero sono una componente essenziale di ogni tentativo di portare avanti il cambiamento trasformativo e performativo di una società sostenibile. I sistemi di pensiero possono superare le debolezze del CT come base concettuale per la progettazione di ICT. Per comprendere adeguatamente la ragioni della sostenibilità, quindi, abbiamo bisogno di capire i sistemi e le loro proprietà emergenti.

¹ Ad esempio, la teoria economica classica è spesso criticata per la sua dipendenza da modelli matematici astratti che catturano e idealizzano comportamenti, piuttosto che ciò che accade realmente nel mondo. Ciò spinge a considerare indicatori globali, come il PIL, che sono sempre più distanti dalla nostra reale percezione del benessere.

Da quanto si è detto prima si comprende che il CT soffre di tre carenze specifiche per le quali i sistemi di pensiero forniscono un toolkit concettuale più appropriato. Queste sono: particolarità della ontologia di dominio, poca apertura al cambiamento, quasi totale assenza di analisi critica. Ad una riflessione attenta si comprende che queste sono il frutto di una visione utilitaristica della formazione. Ma, proprio allo scopo di comprendere meglio la cornice concettuale, scendiamo un po' più nello specifico.

Si è già visto che il CT fornisce un'ontologia dei concetti di calcolo, e una serie di condizioni. Ad esempio, astrazioni procedurali e dati forniscono gli elementi costitutivi di soluzioni computazionali, e la composizione sequenziale e parallela provvedono a curare i modi di metterli insieme. La scomposizione gerarchica viene utilizzata per ridurre i problemi complessi e l'incapsulamento è utile allo scopo di creare soluzioni riutilizzabili. Tuttavia, questa ontologia si focalizza su soluzioni computazionali, di solito a scapito della analisi dettagliata dei problemi e dei contesti problematici. Inoltre, il sottocampo dei requisiti ingegneristici estende l'ontologia dei concetti computazionali (Nuseibeh and Easterbrook, 2000). In particolare, i requisiti ingegneristici aggiungono concetti come parti interessate, gli obiettivi (e la soddisfazione o la negazione di obiettivi), gli scenari, i compiti, e mezzi-fini di analisi. Fornisce anche gli strumenti per la modellazione di situazioni problematiche, come le attività di livello enterprise, e le

interdipendenze tra gli attori in un'organizzazione. Quindi, le soluzioni computazionali vengono raramente valutate su più tempi o da diversi punti di vista. Si avverte pertanto la parzialità e il riduttivismo fenomenologico.

Il pensiero sistemico indirizza questa lacuna attraverso un insieme di concetti più adeguati a comprendere e ragionare sul comportamento. Questi concetti includono:

1. lo stock e i flussi. Il modo più semplice per iniziare una analisi dei sistemi è quello di esplorare in termini di scorte (quantità che variano nel tempo) e flussi (ingressi e uscite di titoli che riguardano la loro livelli). Gli stock possono essere quantitativi concreti, come il livello di acqua in un lago o la migrazione di persone di una città. Oppure, possono essere quantità astratte, come le credenze o desideri, le relazioni sociali, o il benessere.
2. il comportamento emergente. I sistemi tendono ad avere proprietà che non possono essere ricondotti a singoli componenti o gruppi di componenti, ma piuttosto derivare dall'interazione di tali componenti con l'intero sistema. Ad esempio, l'effetto rimbalzo è una proprietà emergente dell'interazione di misure di efficienza energetica con il comportamento economico dell'uomo. Vale a dire, il denaro risparmiato dalle misure di efficienza viene spesso utilizzato per altre attività ad alto consumo energetico(Chitins et alii, 2013). Le persone raramente utilizzano le TIC nel modo in cui i progettisti si aspettano, ma il pensiero computazionale non affronta questo problema. Anche se spesso è

difficile prevedere comportamenti emergenti, indicazioni utili possono essere acquisite abbandonando il riduzionismo, pensando in termini di sistemi adattativi complessi, e la modellazione del loro comportamento con dinamiche non lineari (Arthur et alii, 1997).

3. Anelli di retroazione. La struttura di un sistema costringe i suoi comportamenti. Può sostenere il *rinforzo* degli anelli di retroazione che tendono ad amplificare qualsiasi modifica all'interno del sistema attraverso una catena di causa ed effetto che alla fine fornisce un ulteriore cambiamento nella stessa direzione. Oppure il loro *balancing* operando resistenze al cambiamento, attraverso catene di causa ed effetto che alla fine producono un cambiamento in altra direzione.

Il più grande problema da risolvere per il CT è che non offre alcuno strumento concettuale per ragionare su come il cambiamento avviene nei sistemi complessi. Con il CT si tende ad assumere che il cambiamento avviene attraverso l'innovazione (vale a dire la creazione di una nuova soluzione di calcolo). Spesso si accompagna questa condizione con il presupposto che gli individui, se hanno strumenti migliori per aiutarli a diventare più sostenibili, agiscono il loro contesto.

Dall'altra parte, l'approccio sistemico fornisce una serie di teorie del cambiamento estremamente ricche. Ad esempio, perché alcuni tentativi d'intervenire in un sistema sono più propensi a portare avanti il cambiamento rispetto ad altri? Meadows utilizza i

concetti di stock e flussi per costruire una dettagliata analisi dei punti di leva (Meadows, 1999). In generale, le modifiche che alterano la struttura sottostante dei cicli di retroazione hanno maggiori probabilità di generare cambiamenti nel comportamento a livello di sistema rispetto alla modifica di valori delle variabili all'interno dello stesso. Una ulteriore teoria del cambiamento dei sistemi si basa sulle idee dalla teoria del caos. I sistemi sono spesso osservati per avere modelli stabili di comportamenti che durano nel tempo, indotti dai loro anelli di retroazione. Un sistema può riprendersi dopo un evento catastrofico, se le strutture che lo hanno creato in primo luogo sono ancora supportate. Ad esempio, un bosco ri-cresce dopo un incendio, perché esistono ancora le condizioni che hanno favorito la crescita delle foreste. Questo permette a un sistema di essere resistente di fronte di alcuni tipi di cambiamento (Walker and Salt, 2006). Tuttavia, piccoli cambiamenti possono nel tempo sopraffare questi loop di feedback e guidare il sistema a una nuova serie di comportamenti. Il punto di svolta è indotto da un accumulo di piccoli cambiamenti all'interno del sistema che può essere invisibile agli osservatori esterni. Come con le proprietà emergenti, i punti di non ritorno possono essere facili da identificare dopo il fatto, ma difficili da prevedere (Dai et alii, 2012). Forse la teoria più completa di cambiamento nei sistemi di pensiero è il modello Panarchia (Gundersson and Holling, 2002). Questo modello nasce dalla constatazione che molti sistemi naturali passano attraverso un

ciclo di sfruttamento e di crescita. Quindi da una fase stabile in cui le strutture del sistema diventano sempre più rigide e resistenti al cambiamento, e quindi in definitiva un collasso e una ristrutturazione. La teoria della Panarchia suggerisce che questo ciclo opera a più livelli. Con una progressione di distruzione di creatività che è più lenta su grande scala, più rapida a livelli più ridotti.

Come si è detto vi è poi una terza debolezza del pensiero computazionale: non incoraggia il pensiero critico. Questa mancanza di pensiero critico è evidente nella letteratura tecnica, e ancora di più nelle storie delle TIC nei media, dove ogni nuovo pezzo di tecnologia è descritta in termini di funzionalità immediata. Alcun riguardo è riservato a valutazioni circa il suo impatto sulle divisioni più ampie della società, o sulla vita o sull'ambiente.

La teoria sistemica si presta naturalmente a un approccio critico, perché incoraggia l'idea che qualunque sistema può essere visto come componente in un sistema più grande. Weinberg chiama questo il principio di complementarità (Weinberg, 2001). In teoria, dovrebbe essere possibile per due persone che hanno una vista complementare di un sistema poter conciliare completamente i loro punti di vista. Tuttavia, in pratica, è spesso impossibile fare sufficienti osservazioni di un sistema complesso che permettano di conciliare completamente tutte discrepanze tra i punti di vista. È anche chiaro che i nostri modelli mentali di un sistema limitano

la nostra capacità d'impegnarci con esso. La nostra comprensione di un sistema determina il tipo di osservazioni che siamo propensi a fare di esso. Non soltanto, ma tali osservazioni, a loro volta plasmano la nostra comprensione.

4. La proposta di una ricerca

Quindi il dato è che all'interno della cornice formale tutto appare perfettamente regolato. Ma oggi ha senso sviluppare capacità di problem solving, sia pure attraverso il CT, senza integrarne la dimensione del finding, del setting, del posing e, attraverso queste, sostenere i processi decisionali? Una delle conseguenze più evidenti dell'uso delle nuove tecnologie è proprio la necessità di sviluppare il pensiero strategico, una attitudine a compiere scelte e a prendere decisioni. Questo è particolarmente importante per sostenere l'imprenditività delle persone. Quindi, si ritiene che occorra integrare l'approccio computazionale con una più eccedente visione sistemica². Concludendo, i limiti tracciati dal CT sono da iscriversi sostanzialmente all'interno di una cornice

² In quest'ottica si è attivato un gruppo di ricerca che, intorno a tematiche di confine della pedagogia, sviluppa ricerche sul campo orientate a individuare protocolli di attivazione di competenze. In ogni ricerca si cercano vie di integrazione tra innovazione sociale e tecnologica non perdendo di vista la priorità sociale dell'intervento pedagogico. Una priorità che innova partendo dalla tradizione e dalla cultura propria delle comunità che vengono messe sotto la lente della ricerca. Vi sono molti esempi, ma la matrice culturale è pressoché la stessa: rendere la comunità un ambiente in grado di essere attivato all'innovazione. Si scopre così che, ad esempio, le tecnologie, la computabilità dei problemi devono fare i conti con la singolarità degli universi personali e sociali. E soprattutto che, parafrasando Mozorov, non ci sono soltanto chiodi da piantare, così come non ci sono soltanto martelli da utilizzare. Ciò non significa che il CT non costituisca uno strumento utile, ma per diventare una competenza di base, come vuole Wing, necessita di una espansività che ancora non è stata sufficientemente indagata.

concettuale che ha già avuto eminenti espressioni nella storia del pensiero³.

Far diventare il CT una delle capacità di base dell'apprendimento e legarla alle naturali capacità lettura, scrittura e calcolo rappresenta una riduzione delle potenzialità dei nostri processi di

³Leibniz, nel diciassettesimo secolo, sosteneva di poter escogitare una scrittura universale, mediante la quale fosse possibile eseguire calcoli su ogni genere di argomenti e trovare dimostrazioni come nell'algebra e nell'aritmetica. Ne la "Characteristica universalis" e il "Calculus ratiocinator" afferma che un ragionamento dimostrativo può essere trattato alla stregua di un problema matematico: le premesse corrispondono ai dati iniziali, la conclusione alla soluzione. Come per risolvere un problema si applicano, a partire dai dati iniziali, un certo numero di regole di calcolo per ricavare l'elemento incognito, così nel ragionamento da un gruppo di premesse attraverso regole di deduzione si giunge alla conclusione. Ancora più stupefacente è il fatto che già nel quarto secolo avanti Cristo Platone (nel Filebo) abbia intravisto una estensione del significato del verbo calcolare per comprendere l'attività del pensiero coinvolta nella organizzazione del discorso argomentativo. Platone afferma che tra tutte le scienze il primato spetta alla dialettica, la più alta forma di conoscenza, «in grado di investigare la chiarezza, la precisione, e il massimo grado di verità». Ma, se non sai calcolare non riuscirai a discutere del bene e del male e la tua vita non sarà quella di un uomo, ma quella di un'ostrica o di una medusa". Nei procedimenti di calcolo e nei discorsi argomentativi si passa da un anello all'altro o con regole di calcolo o con regole logiche: da ciò che si sa e si conosce essere vero, si produce "un nuovo" vero. In entrambi i casi occorre disporre prioritariamente di un opportuno linguaggio preciso e rigoroso in cui formalizzare la questione, cioè tradurla in nudi simboli, da trattare con regole di manipolazione che non facciano riferimento al significato. È un dato che ci sono diversi punti di vista da cui le persone muovono per risolvere i problemi. Ad esempio, alcune persone hanno una prospettiva idealista, il che significa che cercano di risolvere il problema ponendosi dall'interno di un quadro d'ideali e valori, di solito di natura religiosa o politica. Altre persone muovono da un punto di vista pratico o pragmatico, evidenziando soltanto i risultati realistici. L'approccio più diffuso è, comunque, la combinazione di un metodo analitico e una prospettiva idealista. Il metodo per prove ed errori è la forma più diffusa di risolvere i problemi, ma anche quella meno efficiente. L'approccio di base è esaminare il problema e cercare qualche idea per risolverlo. Se la prima idea non funziona, provare un'altra idea e così via fino a risolvere il problema. Sarebbe che questa forma di problem solving faccia parte della nostra genetica. Tutti noi lo usiamo in un momento o in un altro. In particolare di fronte a qualche nuova situazione che non comprendiamo pienamente e per la quale non siamo in grado d'individuare uno schema nella nostra memoria a lungo termine. Ma il metodo della prova ed errore di solito è un buon approccio ai problemi in cui il tempo non è un fattore importante. Al contrario non è una buona tecnica per problemi che non consentono una sufficiente ripetibilità dei tentativi o richiedono breve tempo per essere risolti. Uno dei vantaggi più interessanti di questa tecnica è che non richiede come prerequisito una grande mole di conoscenze. Al contrario necessita di grande pazienza, e non è un vero pensiero critico. L'approccio analitico alla soluzione dei problemi è invece l'essenza del pensiero critico. Quando ci avviciniamo a un problema dal punto di vista analitico, pensiamo razionalmente e in modo organizzato. Stiamo applicando criteri solidi per il nostro lavoro, in particolare la logica, la precisione e la chiarezza.

pensiero. Occorre sostenere di più l'approccio analitico e l'utilizzo dell'intuizione, perché è molto meno distorto da giudizi e pregiudizi. L'approccio analitico alla soluzione dei problemi è di solito suddiviso in una serie di passaggi. Alcuni autori suggeriscono quattro fasi, altri fino a 13. Per esempio, il famoso matematico Georg Polya ne ha proposto quattro fasi:

1. capire il problema.
2. escogitare un piano.
3. Effettuare il piano.
4. Guardare indietro (riflettere su ciò che ha funzionato e cosa no).

Polya ha elaborato ognuno dei suoi quattro passi in veri e propri elenchi di azioni specifiche che un potenziale risolutore di problemi dovrebbe prendere in considerazione. John Dewey ha proposto sette punti del problem-solving plan. A quest'ultimo approccio ci ispiriamo in modo diretto. La proposta deweyana alla soluzione dei problemi è induttiva. Spesso è usato in attività che iniziano con una ipotesi, come nei casi della matematica e delle scienze. Se utilizzato con sistematicità e con un po' di pratica, può facilmente diventare una strategia cognitiva di grande efficacia. Intorno a questo modello si possono ricomporre curricula formativi che sviluppino competenze oltre che capacità di calcolo e si può riconoscere un più giusto ruolo al CT.

Riferimenti bibliografici

Arthur W. B., Durlauf S., Lane D., and Lane D. A. (1997). *Introduction: Process and Emergence in the Economy*, in Arthur

W. B., Durlauf S., (Eds). *Addison-Wesley in The Economy as an Evolving Complex System II*. Addison-Wesley: Boston

Chitnis M., Sorrell S., Druckman A., Firth S. K., and Jackson T. (2013). *Turning lights into flights: Estimating direct and indirect rebound effects for UK households*. In *Energy Policy*, vol. 55, pp. 234–250.

Dai L., Vorselen D., Korolev K. S., and Gore J. (2012). *Generic Indicators for Loss of Resilience Before a Tipping Point Leading to Population Collapse*, in *Science*. vol. 336, no. 6085, pp. 1175–1177.

Denning P. J., (2009). *Beyond computational thinking*. *Communications of the ACM*. vol. 52, no. 6, p. 28.

Easterbrook S. M. (1993), *Negotiation and the Role of the Requirements Specification*. in E. Horwood, P. Quintas. *Social Dimensions of Systems Engineering: People, processes, policies and software development*. Ellis Horwood: London. pp. 144–164.

Goguen J. A. (1992). *The Dry and the Wet*. in E. Falkenberg, C. Rolland, and E.-S. El-Sayed. *Information Systems Concepts*. Elsevier: North Holland, pp. 1–17.

Gundersson L. and Holling C. S. (2002). *Panarchy: Understanding Transformations In Human And Natural Systems*. Island Press: Washington.

Holling C. S. (2001). *Understanding the Complexity of Economic, Ecological, and Social Systems*. in *Ecosystems*, vol. 4, no. 5, pp. 390–405.

Jones E. (2011). *The Trouble with Computational Thinking*. Available:<http://csta.acm.org/Curriculum/sub/CurrFiles/JonesCTOnePager.pdf>.

Manson S. M. (2001). *Simplifying complexity: a review of complexity theory*. in *Geoforum*, vol. 32, no. 3, pp. 405–414.

Maslow A. (1966). *The psychology of science: A reconnaissance*. Maurice Bassett Publishing:

Meadows D. H. (1999). *Leverage Points: Places to Intervene in a System*. The Sustainability Institute: Lynedoch.

Morozov E. (2013). *To Save Everything, Click Here: The Folly of Technological Solutionism*. PublicAffairs: New York.

Nuseibeh B., Easterbrook S. M. (2000). *Requirements Engineering: A Roadmap*, in Finkelstein A. C. W. Limerick. *The Future of Software Engineering*. Companion volume to the proceedings of the 22nd IEEE/ACM International Conference on Software Engineering (ICSE2000). IEEE Computer Society Press: Ireland, pp. 35–46.

Rittel H. W. J., Webber M. M. (1973) *Dilemmas in a General Theory of Planning*. Policy Sciences, vol. 4, pp. 155–169.

Walker B. H.,D. Salt (2006). *Resilience thinking: sustaining ecosystems and people in a changing world*. Island Press: Washington.

Weinberg G. M. (2001). *An Introduction to General Systems Theory*. Dorset House: London.

Wing J. (2006). *Computational Thinking*. Communications of the ACM, vol. 49, no. 3, pp. 33–35.