MATEMATIIKKA JA TIETOTEKNIIKKA

Pekka Orponen
Jyväskylän yliopisto, matematiikan laitos

Tarkoitukseni on seuraavassa, matematiikan ja tietotekniikan monimuotoisia vuorovaikutussuhteita tarkastelevassa katsauksessani muistuttaa joistakin näiden oppiaineiden historiallisista ja tieteellisistä yhteyksistä, jotka nykyisessä tietotekniikkakeskustelussa pyrkivät mielestäni aiheettomasti unohtumaan. Kirjoituksen lopussa esitän joitakin huomioita myös tietotekniikan suhteesta varsinaisiin luonnontieteisiin — biologiaan, kemiaan ja fysiikkaan.

Matematiikan ja tietotekniikan eli, kuten eri aikoina hieman eri painotuksin on sanottu, tietojenkäsittelyopin tai tietojenkäsittelytieteen erityissuhde juontaa jo aivan tietokoneiden kehityksen alkuajoilta — olivathan monet 1940-luvun tietotekniikan suuret pioneerit aikansa etevimpiä matemaatikoita. Nimekkäimmät esimerkit ovat tietenkin sovelletun matematiikan unkarilais-amerikkalainen universaalinero John von Neumann ja brittiläinen Alan Turing, joka tunnetaan parhaiten loogikkona, mutta jonka tutkimukset itse asiassa kattoivat melko laajan alan matemaattisesta tilastotieteestä teoreettiseen biologiaan. Suomessakin kotimaisen tietotekniikan tutkimuksen avasi akateemikko Rolf Nevanlinnan 1950-luvulla johtama Matematiikkakonekomitea [5], ja onpa Kansainvälisen matemaattisen unionin joka neljäs vuosi jakama tietojenkäsittelyteorian palkintokin nimetty Nevanlinnan mukaan, vaikka Nevanlinnan oma tieteellinen kiinnostus ei koskaan tietojenkäsittelyteoriaan sanan nykyisessä merkityksessä ulottunutkaan.

Edes visionäärit von Neumann ja Turing, joiden monet tietojenkäsittelyteoreettiset ideat edelleen hämmästyttävät syvyydellään ja pysyvällä ajankohtaisuudellaan, eivät kuitenkaan näytä täysin oivaltaneen sitä, minkälainen vaikutus tietotekniikalla tulee olemaan yhteiskunnan arkipäivään — tai tarkemmin sanoen sitä, minkälaisen teknisen kehityksen kautta tietokoneiden tarjoamat periaatteelliset mahdollisuudet tulevat toteutumaan.

Esimerkiksi ensimmäisen nykyaikaisen ohjelmointikielen, numeerisessa tietojenkäsittelyssä valta-asemansa säilyttäneenkin FORTRANin kehittäjä John Backus muistelee [31, s. 11], että kun John von Neumann 1950-luvun alussa parin vuoden ajan ennen kuolemaansa toimi IBM-yhtiön konsulttina, hän oli suositellut Backuksen FORTRAN-projektin lakkauttamista hyödyttömänä rahareikänä. Von Neumannin mielestä tietojenkäsittelyn todelliset ongelmat liittyivät nimittäin laskenta-algoritmien kehittämiseen ja niiden numeeriseen vakauteen. Algoritmien kääntäminen tietokoneen konekielelle, siis asianmukaisiksi ykkösten ja nollien jonoiksi, on vain työläs mutta sinänsä triviaali tuotantovaihe, jonka toteuttamiseen ei tarvita kalliita tietokoneita, vaan se voidaan hoitaa melko vähän koulutetulla, halvalla ihmistyövoi-

malla. Tämähän oli 1950-luvun alun tilanteessa sinänsä totta, mutta ilman automaattisesti käännettäviä korkean tason ohjelmointikieliä ei nykytilanteeseen johtanut tietotekniikan läpimurto olisi voinut tapahtua, ja Backuksen FORTRAN-projekti oli siten nykyisestä näkökulmasta tarkastellen vallankumouksellinen.

Myös Alan Turingin kerrotaan [18, ss. 398–399] vähätelleen ohjelmoinnissa käytettävien formalismien merkitystä osallistuessaan Manchesterin yliopiston Mark I -tietokoneprojektiin 1940-luvun lopulla. Kun Turing huomasi, että koneen katodisädeputkelle ilmaantuvat laskutulokset oli helpointa tulkita tavanomaisten kymmenjärjestelmän lukujen sijaan 32-järjestelmän lukuina, hän siirtyi käyttämään tätä esitystä kaikessa projektiin liittyvässä työssään — mukaanlukien projektipalaverit muiden tätä innovaatiota kummastelleiden tutkijoiden kanssa. Turingin esitysten seuraamista vaikeutti vielä entisestään se, että hän oli todennnut helpoimmaksi käsitellä 32-kantaisia lukujaan käännetyssä numerojärjestyksessä, siis vähiten merkitsevä numero ensin. Mark I -tietokoneen syöttö- ja tulostusformalismin kehittämistä Turing piti "pelkkänä paperityönä" ja "tärkeilevinä pikku yksityiskohtina".

Näin jälkikäteen tarkastellen on mielenkiintoista havaita, että tietotekniikan koko yhteiskunnan läpäisevä yleistyminen on perustunut — nopean laitekehityksen ohella — juuri huomion kiinnittämiseen noihin von Neumannin ja Turingin vähättelemiin "triviaaliin käännöstyöhön" ja "tärkeileviin yksityiskohtiin", siis entistä parempien korkean tason ohjelmointikielten ja -menetelmien kehittämiseen. Tehtävä on vaikea ja vaatimukset kovat, ja niinpä onkin jo parikymmentä vuotta puhuttu "ohjelmistotuotannon kriisistä" ja haettu siihen ratkaisua milloin "käyttäjänläheisestä" COBOL-kielestä, milloin "neljännen sukupolven sovelluskehittimistä" tai "oliopohjaisista suunnittelu- ja ohjelmointimenetelmistä". Vaikeus tällä alalla ei ole niinkään asioiden matemaattis-teknisessä puolessa, joka hallitaan nykyisin jo jokseenkin täydellisesti [2], vaan siinä että on kerta kaikkiaan vaikeata suunnitella työkaluja ja yhteistyömuotoja, joiden avulla ihmiset pystyisivät tehokkaasti ja mahdollisimman virheettömästi rakentamaan ja ylläpitämään suunnattoman mutkikkaita ohjelmistokokonaisuuksia. Matemaatikoiden von Neumannin ja Turingin monessa muussa suhteessa erittäin avariin tulevaisuuden visioihin ei siten mahtunut se, miten suuriksi ja monimutkaisiksi ohjelmistoprojektit tulevat kasvamaan.

Voi silti olla, että näiden pioneerien nyt huvittavalta vaikuttavassa suhtautumisessa ohjelmointiformalismien yksityiskohtiin oli hiven perustaakin. Nykyisin, kun kovin paljon huomiota kiinnitetään niin julkisuudessa, teollisuudessa kuin yliopistoissakin ohjelmistotuotannon uusimpiin menetelmiin ja virtauksiin, jää helposti vähemmälle huomiolle se merkittävä, mutta vaikeammin ymmärrettävä edistys, jota tietotekniikan matemaattisessa teoriassa jatkuvasti tapahtuu, ja joka muodostaa tämän tekniikan pitkän aikavälin kehityksen todellisen perustan. Samoin jää huomaamatta se, minkälaisia matemaattisia valmiuksia tämän tekniikan innovatiivinen soveltaminen esi-

merkiksi teolliseen tuotantoon vaatii. Tämä näkökulman yksipuolistuminen on alkanut mielestäni suorastaan haitata tietotekniikan yliopisto-opetusta ja sitä kautta edellytyksiämme hyötyä tästä tekniikasta tehokkaimmalla mahdollisella tavalla. Palaan tähän kysymykseen hieman tuonnempana, kun olen ensin hieman tarkastellut tietotekniikan, tai tietojenkäsittelyopin historiaa ja luonnetta akateemisena oppiaineena.

Tietojenkäsittelyä opetettiin 1960-luvun alkuun asti yliopistoissa sovelletun matematiikan ja fysiikan erikoiskursseilla, sekä erikseen perustettujen laskentakeskusten käyttäjäkursseilla [21]. Ensimmäinen erityinen tietojenkäsittelyopin laitos perustettiin Yhdysvalloissa Purduen yliopistoon vuonna 1962, ja Suomessakin seurattiin melko pian perässä: tietojenkäsittelyopin koulutus aloitettiin Tampereen yliopistossa vuonna 1965 Reino Kurki-Suonion, ja Helsingin yliopistossa vuonna 1967 Martti Tienarin johdolla. Jyväskylän yliopiston ensimmäinen oppituoli perustettiin samoin vuonna 1967 yhteiskuntatieteelliseen tiedekuntaan, ja pari vuotta myöhemmin seurasi toinen matemaattis-luonnontieteelliseen tiedekuntaan. Jälkimmäiseen virkaan nimitettiin vuonna 1972 nykyinen emeritusprofessori Aarni Perko.

Tietojenkäsittelyoppi oli alkuvuosinaan oppiaineena ymmärrettävästi melkoisen köykäinen ja vakiintuneiden, monisatavuotiset perinteet omaavien naapuritieteiden epäluulon kohteena. Oppiaineen sisäisen logiikan ja käytännön tarpeiden johtamana, ehkä myös oppiaineen itsenäisyyden korostamiseksi, valittiin uusilla tietojenkäsittelyopin laitoksilla tutkimussuunniksi sellaisia melko puhtaasti ohjelmistotuotannollisia aloja kuin ohjelmointikielten kääntäjätekniikka tai tietokoneiden käyttöjärjestelmät. Aikaa myöten nämä tutkimussuunnat sitten vakiinnuttivat asemansa "oikean" tietojenkäsittelyopin esikuvina jopa siinä määrin, että aivan viime vuosiin asti on muunlaisia, erityisesti liiaksi sovellettuun matematiikkaan viittaavia tutkimusaiheita joillakin laitoksilla pyritty jopa aktiivisesti torjumaan. Minusta tämä oman reviirin paaluttaminen on ollut outoa, ja alan pitkän aikavälin tieteelliselle kehitykselle suorastaan haitallista, mutta toisaalta ymmärrettävää oppiaineen historiallista taustaa vasten. Yliopistoissa, joissa tietotekniikan opetus on kehittynyt läheisessä yhteydessä matematiikkaan, kuten esimerkiksi Jyväskylässä ja Turussa, ei tällaista eristymispyrkimystä onneksi liene esiintynyt.

Kolmenkymmenen vuoden aikuistumiskautensa kuluessa tietojenkäsittelyoppi, tai kuten nykyisin sanotaan tietojenkäsittelytiede, on kuitenkin vahvistunut oppiaineena siinä määrin, että sen on nyt mahdollista hakea yhteyksiä vakiintuneempiin eksakteihin tieteisiin, erityisesti matematiikkaan mutta myös esimerkiksi fysiikkaan, kemiaan ja biologiaan, aiempaa tasaveroisemmalta pohjalta — jopa niin, että tietojenkäsittelytutkimuksella on uutta annettavaa näille "vanhoille tieteille".

Mainitsen muutaman esimerkin matematiikan, tietotekniikan ja muiden eksaktien tieteiden hedelmällisestä vuorovaikutuksesta viime vuosilta.

Tavallisimmin kun puhutaan matematiikan ja tietotekniikan yhteyksis-

tä, ajatellaan ensimmäisenä tietokoneen käyttöä sovelletun matematiikan numeerisena työjuhtana [28] — tämähän oli historiallisestikin ensimmäisten tietokoneiden alkuperäinen käyttötarkoitus. Numeerisen tietojenkäsittelyn käytännön merkitys onkin huomattava: tietokoneiden kehittyminen on tehnyt mahdolliseksi niin yksityiskohtaisten luonnontieteellisten ja teollisen tuotannon mallien käsittelyn, että on alettu puhua erityisestä "laskennallisesta tieteestä", jossa kallis ja aikaavievä kokeellinen työ ja prototyyppien valmistaminen korvataan niin pitkälle kuin mahdollista tietokonesimulaatioilla. Vaikka numeeriset mallit eivät koskaan voi täydellisesti korvata todellisia kokeita, voidaan niiden avulla saavuttaa huomattavia rahan ja ajan säästöjä.

Tieteellisen laskennan vaatiman numeerisen analyysin lisäksi sovelletaan nykyisessä tietotekniikassa mitä monipuolisimmin myös muiden matematiikan alojen menetelmiä, ja sovellusten kirjo ja vaativuus kasvavat koko ajan. Melko lähellä perinteistä numeriikkaa on vielä *tietokonegrafiikka* [17], josta koneiden ja graafisten ohjelmistojen ylitettyä tietyn käytettävyyskynnyksen on tullut huomattava tuotannonala, ja myös yksi yliopistojen tietotekniikan opiskelijoiden suosikkiaiheista.

Aivan toisenlaisten matemaattisten menetelmien tarpeen on puolestaan nostanut esiin tietoverkkojen käytön viimeaikainen voimakas kasvu. Verkkojen laajamittainen käyttö kaupan ja hallinnon välineenä edellyttää nimittäin välttämättä, että niissä siirrettävä tieto voidaan sekä suojata asiaankuulumattomien urkinta- ja muutosyrityksiä vastaan että autentisoida, so. varmistaa viestien lähettäjätietojen aitous. Näihin tarkoituksiin voidaan soveltaa viimeisten 20 vuoden aikana kehitettyjä, tiettyihin lukuteorian ja ns. laskennan kompleksisuusteorian matemaattisiin tuloksiin perustuvia *julkisen avaimen salausmenetelmiä* [29, 30].

Yksi lehdistössäkin usein esillä ollut salausmenetelmien sovellus on "sähköraha", jossa tietokoneelle tai erilliselle rahakortille voidaan ladata käytöltään käteistä rahaa vastaavaa salattua informaatiota. Toisin kuin asian esittelystä julkisuudessa ehkä voisi päätellä, sähköraha ei ole erään suuren suomalaisen liikepankin keksintö, eikä edes erään tunnetun Internet-palveluntarjoajan luomus, vaan hollantilaisen matemaatikon ja tietojenkäsittelyteoreetikon David Chaumin tutkimustyön [9, 10, 11] tulos, joka edelleen perustuu laajaan sovelletun lukuteorian ja matemaattisen kryptologian kirjallisuuteen.

Matemaattiselta perustaltaan jälleen aivan erilainen tietotekniikan laji on *hahmontunnistus* [13], siis kuvien, äänten, käsialojen yms. "luonnollisten" datalähteiden automaattinen tulkinta. Hahmontunnistusmenetelmien merkitys tietotekniikassa on vahvasti kasvamassa juuri nyt, kun monissa sovelluksissa yritetään siirtyä tietokoneen toimintasääntöjen jäykästä ennaltaohjelmoinnista niiden "älykkääseen" mukauttamiseen kulloistakin toimintaympäristöä vastaaviksi.

Yksi viime vuosina sekä julkisuudessa, soveltajien keskuudessa että yliopisto-opiskelijoiden joukossa paljon kiinnostusta herättänyt hahmontunnistusme-

netelmien perhe ovat ns. *neuroverkot* [7, 16, 20]. Monien neuroverkkomalleista niiden biologisperäisen lähtökohdan takia innostuneen yllätykseksi mallien syvällisempi ymmärtäminen, samoin kuin hahmontunnistustutkimus yleensäkin, edellyttää sekä kohtuullista matemaattisen tilastotieteen tuntemusta että sovelletun matematiikan perustyökalujen, yhden ja useamman muuttujan differentiaali- ja integraalilaskennan sekä numeeristen menetelmien hyvää hallintaa.

Tässä kohden on ehkä hyvä palata kysymykseen yliopistotasoisen tietotekniikan opetuksen luonteesta. Esittämäni esimerkit ovat toivottavasti valaisseet sitä näkökantaani, että todella uusien tietotekniikan menetelmien kehittäminen ja innovatiivinen soveltaminen vaatii vankkaa matemaattista peruskoulutusta. Yhteiskunnalle hyödyllisiä tietotekniikan ammattilaisia voidaan toki kouluttaa vähemminkin matemaattisin vaatimuksin, mutta silloin koulutus välttämättä rajoittuu, tai ainakin painottuu sellaisten ohjelmistotuotteiden rakentamiseen, jotka perustuvat hyvin ymmärrettyihin "käsikirjamenetelmiin".

Tämä tietotekniikka-aineen matemaattisuus on selvästi yllätys monille opintojaan aloittaville ylioppilaille, jotka ovat koulussa tai harrastuksen piirissä tutustuneet vain valmiiden tietotekniikan sovellusten käyttöön tai yksinkertaisten puhdetyöohjelmien nikkarointiin. Toisaalta koulujen tarjoama tietotekniikkaopetus, joka välttämättä keskittyy useimpien oppilaiden tarvitsemien sovellusohjelmien käyttöön, saattaa karkottaa yliopistollisista tietotekniikkaopinnoista niitä, joille aine itse asiassa parhaiten soveltuisi. Tämän ristiriidan ratkaiseminen on hankalaa: ehkä lupaavin tie olisi kehittää koulujen kerhotoimintaa ja laatia sitä varten sopivaa, haastavaa materiaalia.

Kun nyt olen käsitellyt matemaattisten menetelmien merkitystä tietotekniikalle, mainitsen vielä muutaman esimerkin tietotekniikan merkityksestä matematiikalle, ja oppiaineen yhteyksistä varsinaisiin luonnontieteisiin.

Ehkä selkein esimerkki tietotekniikan annista matematiikalle, matemaattisten tekstinkäsittelyjärjestelmien [19, 22] kehitystä lukuunottamatta, on epälineaaristen dynaamisten systeemien eli ns. "kaaosteorian" [15, 33] tutkimusaktiviteetti 1970- ja 1980-luvuilla. Vaikka tutkimusalan perusteet olikin luonut jo Henri Poincaré vuosisadan alussa, ja sen kysymyksiä tutkinut m.m. Helsingin yliopiston matematiikan professori P. J. Myrberg 1950-luvulla [26], alan laajamittainen tutkimus käynnistyi vasta kun suhteellisen halvat tietokoneet yleistyivät, ja tutkijat saattoivat tietokonesimulaatioista saada kvalitatiivisen kuvan epälineaaristen systeemien kiehtovasta globaalista käyttäytymisestä.

Toinen alue, jossa tietotekniikan vaikutus on selvä, joskaan ei niin helposti yksilöitävissä, on yleinen kombinatorisen matematiikan — verkkoteorian, koodausteorian, enumerointiteorian jne. — renessanssi 1960-luvulta alkaen. Tietokoneiden yleistyminen on kiinnittänyt matemaatikoiden huomion uudelleen äärellisiin struktuureihin, ja joissakin tapauksissa kombinatorisilla tuloksilla on kyetty vastaamaan suoraan käytännön tietojenkäsittelyongel-

miin (ks. esim. [8]).

Mainitsen vielä muutaman tuoreen esimerkin tietotekniikan ja luonnontieteiden rikastuvista yhteyksistä. Viime vuosina on kehitetty lukuisia mielenkiintoisia tietojenkäsittelymenetelmiä, joiden esikuvat on saatu fysiikan, kemian tai biologian teorioista. Tämä innovatiivisten uusien menetelmien etsiminen luonnollisista prosesseista on erittäin mielenkiintoinen tietotekniikan kehityssuunta, ja merkki oppiaineen kypsymisestä etsimään innoitusta omien vakiintuneiden rajojensa ulkopuolelta.

Aiemmin mainitsin jo hahmontunnistuksen neuroverkkomenetelmät, jotka tietenkin ovat biologisten hermoverkkojen motivoimia, vaikka niiden täsmällinen käsittely edellyttääkin matemaattisia tarkasteluja. Biologiasta ovat saaneet innoituksensa myös tällä hetkellä suuren kiinnostuksen kohteena olevat "geneettiset" optimointimenetelmät [6, 25], joissa pyritään jäljittelemään eliöpopulaation kehittymistä risteytysten ja luonnonvalinnan kautta kohti entistä parempia "kuntoisuusfunktion" arvoja. Fysiikasta puolestaan on tietotekniikkaan jo lähes arkipäiväiseen käyttöön vakiintunut kiderakenteiden järjestymistä hitaasti jäähdytettävissä kappaleissa jäljittelevä optimointimenetelmä, "simuloitu jäähdytys" [1]. Kemiallisista menetelmistä mainittakoon kokeilut, joissa massiivisesti rinnakkaisia laskentoja yritetään toteuttaa kemiallisissa liuoksissa yksittäisten biomolekyylien vuorovaikutuksiin perustuvilla "biotietokoneilla". Joitakin huomiota herättäneitä kokeellisia tuloksia on tässä suunnassa jo saavutettu [3], ja sekä odotukset että tutkimusaktiviteetti ovat korkealla [23].

Myönteistä on, että tietojenkäsittelyteoriakin on jo pariin kertaan saattanut tarjota fysiikalle uusia tutkimussuuntia, mikropiirien kehittämisen luonnollisesti edellyttämän laajan materiaalitutkimuksen lisäksi. Viime vuosikymmenen loppupuolella työskenteli melko suuri joukko teoreettisia fyysikoita tiettyjen, fysikaalisia spinlasisysteemejä muistuttavien neuroverkkomallien analyysin parissa [4, 27], ja nyt näyttää fyysikkoyhteisöä kiehtovan ns. *kvanttitietokoneiden* teoria.

Kvanttitietokoneet [24, 35] ovat hypoteettisia laitteita, joissa aineen kvanttitilojen teoreettista samanaikaisuutta käytettäisiin tietojenkäsittelyprosessien massiiviseen rinnakkaistamiseen. Kvanttilaskennan idea on ollut periaatteessa tunnettu jo toistakymmentä vuotta [14, 12], mutta alaa pidettiin täysin marginaalisena, kunnes tietojenkäsittelyteoreetikko Peter Shor vuonna 1994 keksi menetelmän [32] suurten kokonaislukujen nopeaan tekijöihinjakoon kvanttirinnakkaisuutta hyväksi käyttäen. Tämä oli mullistava tulos, koska m.m. tärkeimpien julkisen avaimen salausmenetelmien turvallisuus perustuu oletukseen, että kokonaislukujen tekijöinti on laskennallisesti raskas tehtävä. Shorin keksinnön jälkeen keskustelu kvanttitietokoneiden mahdollisista toteutustavoista on käynyt vilkkaana — monet fyysikot näyttävät tosin olevan myös sitä mieltä, että vaikka kvanttilaskenta ei varsinaisesti luonnonlakien vastaista olekaan, niin toimivan kvanttitietokoneen rakentaminen on käytännön syistä mahdotonta. Mielenkiintoista on silti havaita, että kaiken

tämän fysiikan tutkimusaktiviteetin pontimena on ollut puhtaasti matemaattinen tietojenkäsittelyteorian tulos.

Kuten esimerkit osoittavat, tietotekniikan matemaattinen teoria on edistynyt mittavasti niinä noin kolmenakymmenenä vuotena, joina alaa on yliopistollisena oppiaineena opetettu. Uusia tietojenkäsittelymenetelmiä on kehitetty, ja kehitetään jatkuvasti sekä oppiaineen jo olemassa olevista lähtökohdista että muista tieteistä saatavien ideoiden pohjalta. Uusien ideoiden muokkaaminen tietotekniikan kannalta käyttökelpoiseen muotoon edellyttää kuitenkin aina huolellista, ja joskus hyvinkin vaativaa, matemaattista tai joissakin tapauksissa tilastotieteellistä analyysiä. Siten innovatiivinen tietotekniikan tutkimus ja huippuluokan ammattilaisten koulutus menestyvät parhaiten ympäristössä, jossa tietotekniikka-oppiaine on elävässä vuorovaikutuksessa muiden matemaattisten tieteiden — matematiikan, tilastotieteen ja eksaktien luonnontieteiden kanssa.

Viitteet

[1] Aarts, E., Korst, J. *Simulated Annealing and Boltzmann Machines: A Stochastic Approach to Combinatorial Optimization and Neural Computing.* J. Wiley & Sons, Chichester 1989.

[2] Aho, A., Sethi, R., Ullman, J. *Compilers: Principles, Techniques, and Tools.* Addison-Wesley, Reading, MA, 1986.

[3] Adleman, L. Molecular computation of solutions to combinatorial problems. *Science 266* (11 Nov. 1994), 1021–1024.

[4] Amit, D. *Modeling Brain Function: The World of Attractor Neural Networks.* Cambridge Univ. Press 1989.

[5] Andersin, H., Carlsson, T. ESKO – ensimmäinen suomalainen tietokone. Ss. 11–23 teoksessa [34].

[6] Bäck, T. *Evolutionary Algorithms in Theory and Practice.* Oxford Univ. Press 1996.

[7] Bishop, C. *Neural Networks for Pattern Recognition.* Oxford Univ. Press 1995.

[8] Bollobás, B., Chung, F. *Probabilistic Combinatorics and Its Applications.* Proc. Symposia in Applied Mathematics 44, American Mathematical Society, Providence, RI, 1992.

[9] Chaum, D. Security without identification: Transaction systems to make the big brother obsolete. *Communications of the ACM 28* (1985), 1030–1044.

[10] Chaum, D. The dining cryptographers problem: Unconditional sender and recipient untraceability. *J. of Cryptology 1* (1988), 65–75.

[11] Chaum, D. Achieving electronic privacy. *Scientific American*, Aug. 1992, 96–101.

[12] Deutsch, D. Quantum theory, the Church-Turing principle, and the universal quantum computer. *Proc. R. Soc. Lond. A 400* (1985), 97–117.

[13] Duda, R., Hart, P. *Pattern Classification and Scene Analysis.* J. Wiley & Sons, New York, NY, 1973.

[14] Feynman, R. Simulating physics with computers. *Internat. J. of Theoretical Physics 21* (1982), 467–488.

[15] Gleick, J. *Chaos: Making a New Science.* Viking, New York, NY, 1987.

[16] Haykin, S. *Neural Networks: A Comprehensive Foundation.* IEEE Press, New York, NY, 1994.

[17] Hearn, D., Baker, M. *Computer Graphics.* Prentice-Hall, Englewood Cliffs, NJ, 1994.

[18] Hodges, A. *Alan Turing: The Enigma.* Simon & Schuster, New York, NY, 1983.

[19] Knuth, D. *The TEXbook.* Addison-Wesley, Reading, MA, 1986.

[20] Kohonen, T. *Self-Organizing Maps.* Springer-Verlag, Berlin 1995.

[21] Kurki-Suonio, R. Tietojenkäsittelyopin korkeakouluopetuksen käynnistyminen. Ss. 24–47 teoksessa [34].

[22] Lamport, L. *LATEX: A Document Preparation System, 2nd Ed.* Addison-Wesley, Reading, MA, 1994.

[23] Lipton, R., Baum, E. (eds.), *DNA Based Computers.* DIMACS Series in Discrete Mathematics and Theoretical Computer Science 27, American Mathematical Society, Providence, RI, 1996.

[24] Lloyd, S. Quantum-mechanical computers. *Scientific American*, Oct. 1995, 44–50.

[25] Mitchell, M. *An Introduction to Genetic Algorithms.* The MIT Press, Cambridge, MA, 1996.

[26] Myrberg, P. Iteration von Quadratwurzeloperationen. *Annales Acad. Sci. Fennicae Ser. A I Math. 259:1* (1958).

[27] Peretto, P. *An Introduction to the Modeling of Neural Networks.* Cambridge Univ. Press 1992.

[28] Press, W., Teukolsky, S., Vetterling, W., Flannery, B. *Numerical Recipes in C: The Art of Scientific Computing, 2nd Ed.* Cambridge Univ. Press 1992.

[29] Salomaa, A. *Public-Key Cryptography.* Springer-Verlag, Berlin 1990.

[30] Schneier, B. *Applied Cryptography, 2nd Edition: Protocols, Algorithms, and Source Code in C.* J. Wiley & Sons, New York, NY, 1996.

[31] Shasha, D., Lazere, C. *Out of Their Minds: The Lives and Discoveries of 15 Great Computer Scientists.* Springer Copernicus, New York, NY, 1995.

[32] Shor, P. Algorithms for quantum computation: discrete logarithms and factoring. *Proc. 35th Ann. IEEE Symp. on Foundations of Computer Science (Santa Fe, NM, Nov. 1994)*, 124–134. IEEE Computer Society Press, 1994.

[33] Strogatz, S. *Nonlinear Dynamics and Chaos.* Addison-Wesley, Reading, MA, 1994.

[34] Tienari, M. (toim.) *Tietotekniikan alkuvuodet Suomessa.* Gummerus Kirjapaino Oy, Jyväskylä 1993.

[35] Törmä, P., Suominen, K.-A. Kvanttitietokoneet: teoria ja käytäntö. *Arkhimedes 47* (1995), 271–288.

(Virkaanastujaisesitelmä Jyväskylän yliopistossa 27.11.1996)