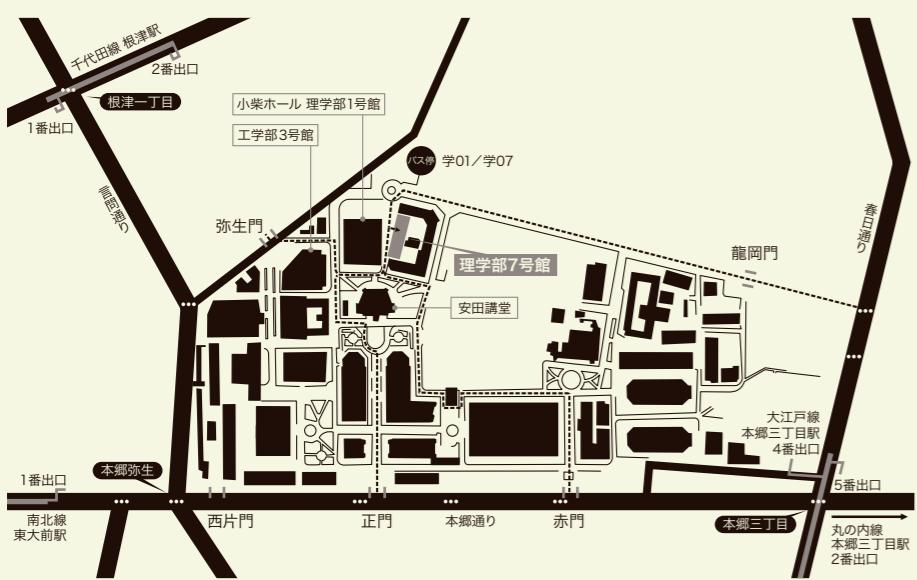


DEPARTMENT OF INFORMATION SCIENCE
2024



東京大学理学部 情報科学科

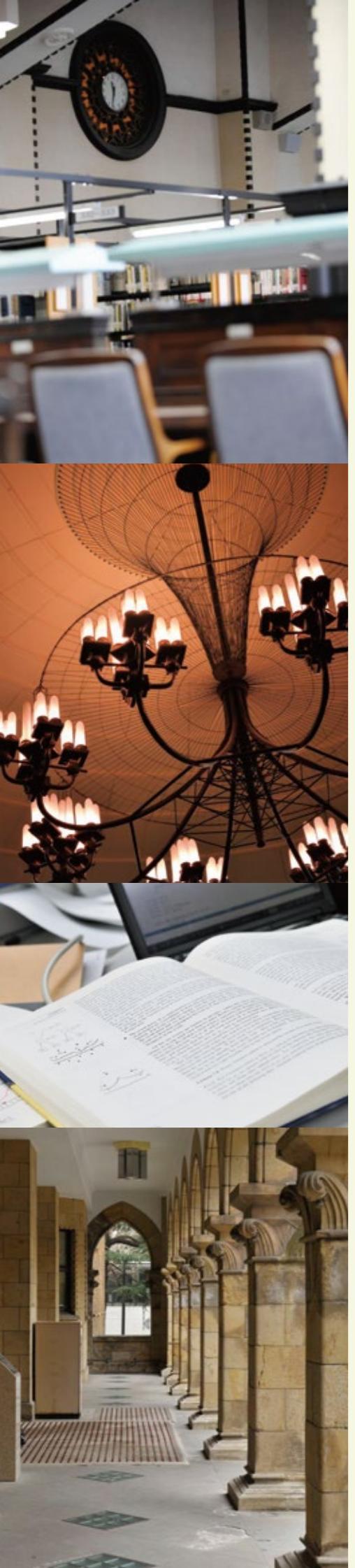
〒113-8656

東京都文京区本郷7-3-1 理学部7号館
Tel. 03-5841-4111/4112

e-mail: www-admin@is.s.u-tokyo.ac.jp

www.is.s.u-tokyo.ac.jp

東京大学理学部 情報科学科



情報科学を通じて未来の社会を設計する

深層学習や大規模言語モデル、生成AIといった人工知能技術が社会を席捲しつつある。産業界では自動運転技術が交通の効率向上と事故の削減を期待され、ロボットや自動化システムはすでに高度なタスクをこなして生産性を向上させている。医療領域では、画像解析や遺伝子解析による疾患の早期発見、治療の個別化が期待されている。大規模言語モデルを応用した文書の自動要約や分析は、ますます広範な分野の業務を効率化していくだろう。

学術研究でも、人工知能技術が急発展した影響は顕著である。これまで難しいとされていた問題が次々に解かれる一方で、これまでとまったく異なる問題が研究の対象として現れてきている。例えば、巨大なブラックボックスと化した人工知能システムをどのように解釈し、どのように制御していくのか。これは、人間の知能の本質にもかかわる重要で根源的な問題である。このような状況を背景に学術研究を行っていくためには、これまで培ってきた知識や技術とはまったく異なる新しい考え方やアプローチが必要だ。

このような変革の時代は、これから学び、世の中にでていく若い皆さんにとっては大きなチャンスである。やれること、やるべきことが、目の前に無限に広がっている。成熟した学問分野においては、その長い歴史の中で培われた知識が膨大にあり、それらを吸収・咀嚼したうえで、先人たちとは違う新しいことを見つけなければならない。そこに大きな困難がある。しかしまったく新しい分野には、そもそもそのような過去の遺産が存在しないわけであり、過去の研究に捕らわれることなく新しいことにチャレンジしていく。

著名な情報科学の研究者一人Alan Kayの言葉に、“The best way to predict the future is to invent it. (未来を予測する最良の方法は、それを創ることだ。)” というものがある。未来を決定していくのは自分達だ、というある情報技術の黎明期を牽引していた先人の意気込みが感じられる言葉である。実際に彼らの開発した情報技術、インターネットやグラフィカルユーザーインターフェイスといった技術が今日の我々の生活を支えている。人工知能技術が急速に発展して、未来の社会のあるべき姿が改めて模索されている今、まさにそのような気概をもって未来の社会を設計していく力が求められている。

情報科学科は、このような社会変革の中核にある人工知能技術および社会を支えている根幹技術である情報技術を、基礎理論から応用まで学ぶことのできる学科である。プログラミング技術といった実務のための技術自体は、働き始めてから身につけることができるかもしれない。しかし、真に社会に変革をもたらすようなスケールの大きい仕事をするためには、情報科学に対する深い理解と経験が不可欠である。当学科では、計算機のハードウェアからソフトウェア、アプリケーションまでを総合的に学び、実験や演習を通じて深く理解して身につけることができる。ぜひ当学科において情報科学を学び、新しい時代を切り開いていってほしい。

五十嵐 健夫

目 次

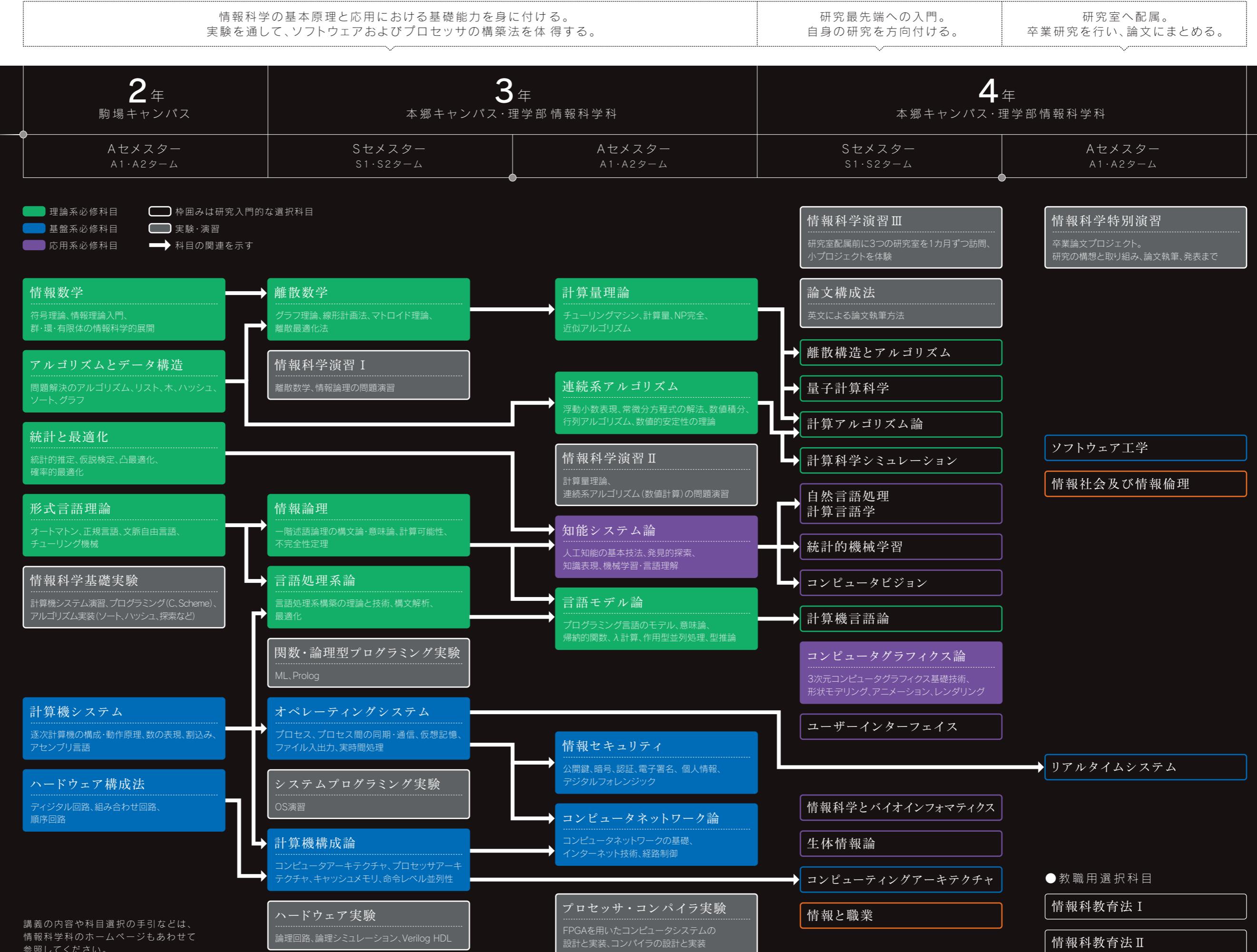
カリキュラムと学習のながれ	4
時間割と進学に必要な履修科目	6
すこし長いFAQ	7
これから2年間、なにを学ぶのか？	
■ 講義・実験・演習をぐるりと紹介	8
■ プロセッサ・コンパイラ実験(CPU実験) 一ほんとうのコンピュータ自作	10
研究ってどんなものだろう？	
演習III—研究室めぐりから卒論への道	12
卒業後の進路	14
情報科学科の研究室紹介	16
■ 小林研究室	17
■ 河原林研究室	18
■ 五十嵐研究室	19
■ 宮尾研究室	20
■ 谷中研究室	21
■ 杉山・石田研究室	22
■ 佐藤研究室	24
■ 横矢研究室	25
■ 馬研究室	26
■ 吉本研究室	27
■ 品川研究室	28
■ 高前田研究室	29

[カバー] チューリング機械と、情報科学科の通称「IS」を重ね合わせた。チューリング機械は、計算を説明する模型のひとつ。単純な規則に従ってテープ上の文字を読み書きする。あらゆる計算を1台の機械で模倣できるという万能性が、現代の計算機の理論的基礎をなしている。

□写真 三浦 健司／高橋 和彦
□文 池野 俊一朗／所司 翼／上田 亮／谷田 直輝／山田 允／村上 直輝／筒井 政成
□デザイン 株式会社ライズ・イメージ・ファクトリー
2024年4月発行

東京大学理学部情報科学科
〒113-8656 文京区本郷 7-3-1 理学部7号館 <http://www.is.s.u-tokyo.ac.jp>
03-5841-4111~4112

カリキュラムと学習のながれ



時間割と進学に必要な履修科目

授業の時間割

●3年Sセメスター(S1・S2)

	月曜日	火曜日	水曜日	木曜日	金曜日
1限 8:30~10:15					
2限 10:25~12:10	オペレーティングシステム	離散数学	情報論理	言語処理系論	計算機構成論
3限 13:00~14:45	システム プログラミング実験	関数・論理型 プログラミング実験		ハードウェア実験	情報科学演習 I
4限 14:55~16:40					
5限 16:50~18:35	情報科教育法 I				

…必修科目 …実験・演習(必修) …選択必修科目 …選択科目

●3年Aセメスター(A1・A2)

	月曜日	火曜日	水曜日	木曜日	金曜日
1限 8:30~10:15					
2限 10:25~12:10	言語モデル論	計算量理論	A1 コンピュータネットワーク論	知能システム論	
3限 13:00~14:45		プロセッサ・コンパイラ実験		プロセッサ・コンパイラ実験	情報科学演習 II
4限 14:55~16:40	連続系アルゴリズム				
5限 16:50~18:35	情報科教育法 II		情報社会及び情報倫理		A2 情報セキュリティ

●4年Sセメスター(S1・S2)

	月曜日	火曜日	水曜日	木曜日	金曜日
1限 8:30~10:15					
2限 10:25~12:10	S1 自然言語処理 S2 ユーザインターフェイス	生体情報論	統計的機械学習	コンピュータグラフィックス論	
3限 13:00~14:45	S1 計算機言語論 S2 計算言語学	情報科学とバイオインフォマティクス			S1 コンピュータビジョン S2 離散構造とアルゴリズム
4限 14:55~16:40		論文構成法			S1 計算科学シミュレーション S2 コンピューティングアーキテクチャ
5限 16:50~18:35	情報科教育法 I				
6限 18:45~20:30		情報と職業			

●4年Aセメスター(A1・A2)

	月曜日	火曜日	水曜日	木曜日	金曜日
1限 8:30~10:15					
2限 10:25~12:10					
3限 13:00~14:45	A1 リアルタイムシステム				
4限 14:55~16:40	A1 ソフトウェア工学				
5限 16:50~18:35	情報科教育法 II		情報社会及び情報倫理		

進学に必要な履修科目(2024年度進学生)

●必修科目

科目番号	科目	単位
0510001	情報数学	2
0510002	形式言語理論	2
0510003	計算機システム	2
0510006	ハードウェア構成法	2
0510007	アルゴリズムとデータ構造	2
0510008	情報科学基礎実験	2
0510009	統計と最適化	2

●選択科目

科目番号	科目	単位
0505001	代数と幾何	4
0505007	代数と幾何演習	2
0505003	集合と位相	4
0505008	集合と位相演習	2
0505005	複素解析学 I	4
0505009	複素解析学 I 演習	2
0515007	物理数学 I	2
0515009	電磁気学 I	2
0515076	物理数学 II	2
0560501	生物情報学基礎論 I	2
0560502	生物情報学基礎論 II	2

教養学部第2学年に左の必修7科目(合計14単位)、ならびに選択科目から4単位以上を学修してください。ただし、これ以外の選択科目(理学部他学科あるいは他学部の第2学年専門科目)を学修している場合でも、所定の期日までに科目認定届を提出し、学科会議の承認を得られれば、選択科目として認められることがあります。

また、4学年での情報科学特別演習の履修のために、第2学年専門科目での必修科目14単位のうち12単位以上をあらかじめ取得しておくことが必要です。

すこし長いFAQ

よく寄せられる情報科学科への問合せにお答えします。

工学部の情報系の学科との違いは何ですか?

理学部のココロとして、「ものごとを根源からとらえる」ということがあります。またその結果、それまでにない、まだ世の中に普及していない題材を扱うこともあるので、ものをイチから作れる力が大切になります。

情報科学科では、コンピュータ分野の先端的な研究を手がけていますが、その前提となるコンピュータの原理や理論的な背景を知り、ものごとを抽象化してとらえる力、また何もないところからモノを作れるだけの技術力を付けることを重視しています。

土台の部分から勉強することは、一見回り道のように思えるかもしれません。けれども、卒業後にどの分野に進んだとしても、この基礎体力が必ず役立つはずです。

プログラミング経験が豊富でないと進学できないでしょうか?

これまでに高校や大学において授業で

学習したこと以上に特別な経験を必要とするはありません。進学時に多少の経験の差があったとしても、十分な指導を行いますので安心してください。3年生の実験では、自分で工夫して作ったものが動くという、プログラミングの面白さを味わえます(課題→p.8)。アルゴリズムを考えるのが好きなら、すぐにとけこめるでしょう。自分で考えたり作ったりするのが好きでないと、苦労するかもしれません。

情報科学科の「実験」とは何ですか?

たとえば物理の実験では、X線や電気、レーザーなどの特性を体感的につかめるよう課題が出され、検証結果を提出したりします。情報科学科では、3年生の2つの実験を通じて、コンピュータの原理を奥底から理解します。

Sセメスターの『システムプログラミング実験』は、馴染みのあるソフトウェアを自分で作ってみるものの、課題に対する解法を考えて設計・プログラミングします(課題→p.8)。

Aセメスターの『プロセッサ・コンパイラ実験』は、与えられたコンピュータグラフィックスのプログラムが動作するように、CPU、コンパイラ、ツールを設計・開発するもの。どのようなCPUやソフトウェアを作るかを考えることから始まります(CPU実験→p.10)。

所定の結果になると実験することなく、問題の設定・設計を自分で考えるクリエイティブな要素があり、実験結果が十人十色になるのが面白いところです。実験から、思わぬ発展に結びつくこともあります。

1限の授業がないのでビックリしました

情報科学科は、結果と使う時間の配分を自分でデザインする、自由な雰囲気の学科です。1限の時間を課題や自分の勉強に使う人も、休養に充てる人もいます。また、選択する分野にもよりますが、4年生に進んでからの研究題材も自由です。学科内で

は、個人が多様性のある活動をしつつも、ワークスペースなどでよく協調している光景が見られます。少人数教育の学科なので、このような運営が可能になっています。

1限の授業がなくて驚かれる方もいます。が、決して「ゆるい」わけではありません。

数学と情報科学の関連性について教えてください

コンピュータは数学や論理学と密接なつながりをもって発展してきました。情報科学科では、情報論理、コンピュータのさまざまな部分で応用されている離散的な理論分野、計算量理論に力をいっています。

地図の塗り分け問題がレジスタ(CPU内にある数値を格納するための回路)割り当てやプリント基板の配線検査時間の短縮に使われていることをご存じの方もあるでしょう。グラフを考えれば、地図の問題がレジスタやほかのいろいろな問題に結びついで解けるのが面白いところです。

ものごとにどのような複雑性があり、どういう問題に帰着するかは、モノづくりに限らず直感的に知っていないといけない知識ですが、これらは計算量理論で身に付けることができます。

現在の知能システムを支える機械学習では、データを数理モデルによって抽象化する過程で、線形代数、関数解析、確率統計などを縦横無尽に使います。

また天気予報に使われる物理シミュレーションでは、微分積分の数値計算を効率良く行うアルゴリズムが使われます。

これから2年間、何を学ぶのか?

情報科学科では、ハードウェアのような基盤層からアプリケーションのような上位層まで、講義で理論や動作原理、設計思想を学び、実験で実際に作ってみることにより、コンピュータシステムの全体像を深く理解できるようになっています。ここでは、実験科目とそこで作成するプログラムの一部を紹介します。日ごろ使っているコンピュータシステムは、いったいどのような仕組みで動いているのでしょうか。

ハードウェア

「ハードウェア実験」や「CPU実験」では、CPUを始めとするハードウェアを自分の手で設計し、その仕組みを学びます。電子回路に触れ、加算器や簡単なCPUを作成し、最終的にはチームでCPUを設計・制作します。チーム対抗で速さを競い合うのが慣例になっています。

OS・基礎ソフトウェア

コンピュータにおけるOSの役目は、アプリケーションがハードウェア資源を効率よく使えるようにすること。OSの授業では、多数のアプリケーションを同時に動かすためのCPUスケジューリング、メモリの仮想化などを学びます。

「システムプログラミング実験」の前半では、システムコールという仕組みでOSの中核機能を直接呼び出し、マルチスレッドプログラミングや、ソケット通信を体験、最後にシェルを作成します。シェルというのは、ユーザーのコマンド入力からOSにプログラムを実行させる、ユーザーとOSの間を仲介するプログラムです。一見簡単そうに思えますが、実際に作ってみるとさまざまな困難に直面します。問題をクラスメートと協力しながら解決し、OSに対する

理解を深めています。

実験後半では、OSのない環境でハードウェアを直接制御し、簡単なOSを作ることを目指しペアメタルプログラミングを行います。シェルの作成よりもさらにたいへんですが、ハードウェアの力を引き出す用途にも役立てられます。

言語処理系論・コンパイラ実験

プログラムをコンピュータ上で実行するためには、コンパイラという翻訳プログラムによりCPUの命令(機械語)へ変換する必要があります。「言語処理系論」の講義では、そんなコンパイラを構築するための理論や技術について学びます。

実験では、OCamlという関数型プログラミング言語を用いて実際にコンパイラを自分の手で作成、その仕組みを深く理解します。また、ただプログラムを各自で定義した機械語に翻訳するだけではなく、各種の最適化手法を採り入れて翻訳後の機械語コードが高速に動作するよう改良していきます。チームメートと実行命令数や実行時間を競い合うのも楽しみです。

数値計算

現代の物理学を支える物理シミュレー

ションは、微分方程式に従う連続量の数値計算によって実現しています。数学的には解くことが難しい微分方程式も、コンピュータで計算を繰り返すことで近似的に計算が可能になります。講義では誤差が広がらないように上手く計算するためのアルゴリズムを学び、演習で常微分方程式に対するRunge-Kutta法などを実装します。

演習では高速化を目指し、プロセス間通信やGPUを活用して、並列計算にも挑みます。並列計算は計算量が増す一方の現代に必要不可欠ながら、専門課程までは触れる機会の少ない分野です。並列計算の工夫によって性能が大きく向上していくさま、またその実現の難しさを実感できる演習は刺激的です。

知能システム

「人工知能とは何か?」という問い合わせから始まり、数理最適化や統計的機械学習、強化学習、言語処理、そして近年注目を集めているニューラルネットワークなど、人工知能に関する基礎的な手法を学びます。講義で理論や仕組みを学んだあと、実際に自分の手で手法を実装して動かし、さらに理解を深めます。知能のモデル化というのは簡単にできるものではなく、知能とは何なのかを改めて考えさせられます。

ユーザーインターフェイス

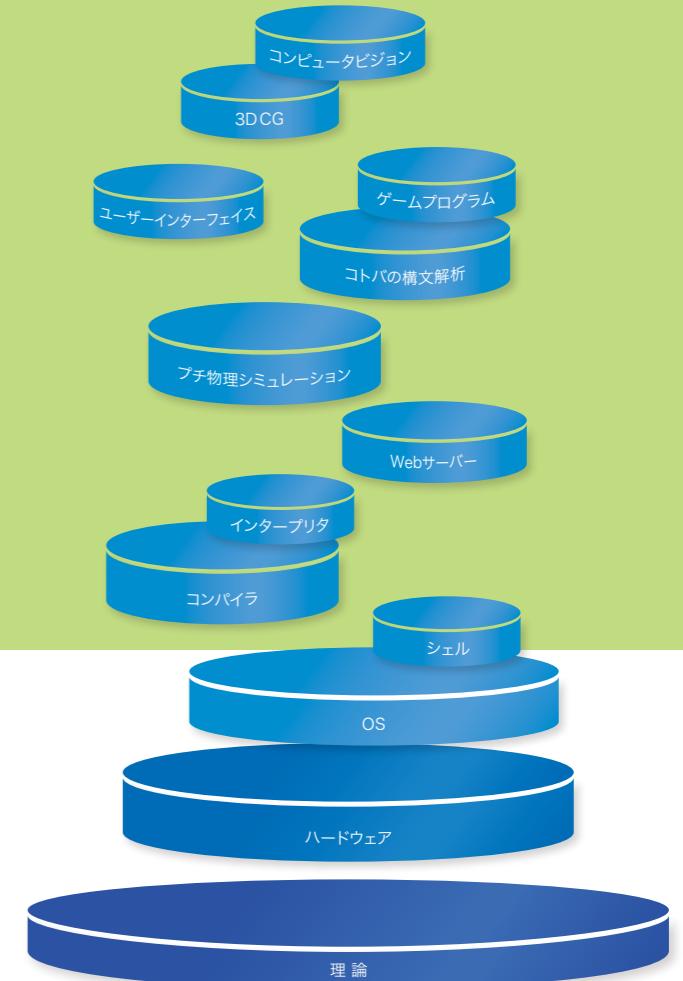
ユーザーインターフェースは、「コンピュータを人間が使用する」ことに焦点を当てた分野です。目的を簡単に・効率的に・気持ちよく達成するため、ソフトウェアは機能だけでなく人間の操作性(インターフェース)を工夫する必要があります。

授業では、コンピュータグラフィックスや機械学習などのテーマを取り上げ、直感的な操作を実現するためのインターフェースが紹介されます。また、毎週の講義内容に関連して、「表情を付けた自動運転車」の性能を評価する実験計画の考案、新しい素材の3Dプリントの考案など、面白い練習課題が与えられます。

最終課題は、ウェブ上のアプリケーションを工夫して実装し、実際に使ってもらうプロジェクト。思ったように使ってもらえない様子を見るのは新鮮で、良い経験です。

離散数学

離散数学は、文字通り離散的な(連続でない)対象を扱う数学です。離散数学には順列や組合せを網羅的に調べることで解ける問題があり、解くのにコンピュータを使用できますが、その解法を高効率にする研究が重要です。例えば「都市の集合と各



実験と課題でコンピュータをまるごと作って理解する

情報交換の場となっている ワークスペース

情報科学科の学生にはノートPCが貸し出され、そのPCを使用して実験のプログラムなどを作成します。課題は、大学内などに限らず、好きな場所で。

学科内には学生用の控室が用意され、研究室配属前の学生も自分の席を持てます。コロナ禍の影響により一時閉鎖されていましたが、内装を新しくして再び利用できるようになりました。このワークスペースの存在意義は大きく、課題の情報交換や共同作業の場になっています。

プログラムの作成では、似たような問題で行き詰まることが多い、情報交換の効果は絶大です。TAや先輩が様子を見に来て、指導してくれることもあります。比較的自由な雰囲気で、課題を片付けてボードゲームを楽しむ光景も見られます。

(2022年4月 池野 俊一朗・所司 翼・上田 亮)

CPU実験——ほんとうのコンピュータ自作

3学年のAセメスターになると、『プロセッサ・コンパイラ実験』——通称「CPU実験」が始まり、3~4人の各チームに、FPGA基板と道具がいくつか渡されます。ミッションは「半年かけてできるだけ速いコンピュータを作れ」。それから翌年3月の発表会までに、課題のCGプログラムが動くよう、独自のCPUやコンパイラなどをイチから設計・製作します。ハードなものの、OB・OGの誰もが「楽しかった」と口を揃えるこの実験の様子を、紹介しましょう。

実験は、まずCPUの命令セット/アーキテクチャの設計から始め、CPU、コンパイラ、アセンブラーを分担して実装、というふうに進みます。

みんなでアーキテクトになる

最初に、CPUの命令セット——CPUが備える命令群などのアーキテクチャを決めます。複雑な仕様にすると完成させるのが難しくなるので、最初は既存のCPUアーキテクチャを参考に、シンプルな設計から始めることができます。CPUの実装のしやすさと、コンパイラの開発のしやすさは、往々にして相容れないもの。うまくそのバランスをとることがたいせつです。

仕様が固まると、各自の興味・得意不得意を考慮して分担を決め、開発にとりかかります。実は、技術的な知識だけでなく、

半年にわたるプロジェクトワークもたいへん貴重な経験になります。

CPUを詳細に設計する

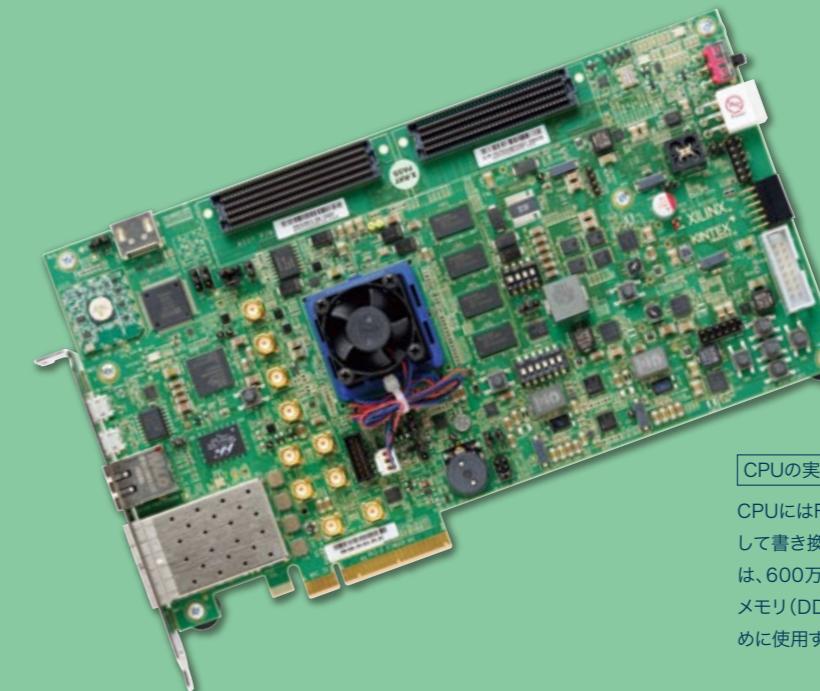
回路が大規模になった現在、論理回路の実装は、回路図上でゲートを配線する代わり、HDL(ハードウェア記述言語)を用いるのが主流です。HDLはプログラミング言語に似ていて、回路の動作を詳細に記述できます。このHDLの記述を、FPGA用の開発ツールを使って回路に合成し、FPGA内に自動配置・配線します。しかし、開発ツールが最適の合成と配置を約束してくれるわけではありません。思い通りの回路になるように、開発ツールの動作を見越してHDLを書くのも腕の見せどころ。設計した論理回路は、開発ツールのHDLシミュレータで表示される波形図で検証します。

ツールいろいろ

CPUが動作したすまで、ソフトウェア係は何もしないのでしょうか？ いいえ。CPUと並行して、CPUシミュレータなどの

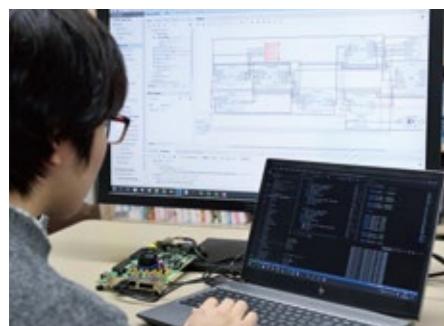
コンパイラを開発する

課題のCGプログラムは、MLというプログラミング言語で書かれているので、そのコンパイラが必要です。最近はMLで実装された洗練されたMLコンパイラがあるのでも、これを改造してまず動作させ、より効率の良い命令列を生成するように改良することが多いようです。既存の最適化手法も採り入れますが、やはり、自分たちのCPUに合わせた独自の方法を試行錯誤することになります。三角関数のような一般的な関数も、コンパイラとともに用意します。



CPUの実装に使用するFPGA評価ボード

CPUにはFPGA(電気的な方法で内部のロジックを自由にデザインして書き換えるLSI)を使用する。写真中央の冷却ファンの下には、600万ゲート規模の回路を実装可能なFPGAがある。周りに、メモリ(DDR4)、電源を供給したりFPGAに論理回路を送りこむために使用するUSBなどの入出力コネクタが備えられている。



モジュールごとにHDLで記述した回路をつなぎ合わせて構成する。



HDLシミュレータが表示する波形図で、設計した論理回路を検証する。



FPGAに論理回路のデータを送り込んでテストする。

ツールも自作します。ツールが設計した命令セットの振る舞いを疑似的に再現してくれるおかげで、コンパイラが生成した命令列を検証したり、実行時間を予測してアーキテクチャを改良したりすることができます。命令の取捨選択のために命令の使用頻度を調べたり、パラメータを変更して実行するためにも使われます。

テスト

ひととおり出来上がると、実際にFPGAに回路のデータを送り込んでテストします。一見うまく動いているようでも、不具合や仕様の解釈違いはあるもので、ここから完成までは思いのほか長くかかります。作ったものが思いどおりに動作しないときには、ロジックアナライザで調べることもあります。



のは、とても気分の良いものです。

発表会には、院生なども大勢訪れます。それぞれのチームが半年かけて練った自分たちのアーキテクチャやコンパイラを熱く語り、無事に完成したチームは実際にCPUを実行させてデモを行います。

年々更新されていく記録には、先輩から継承されたノウハウが数値となって表されています。2016年にFPGAの規模が拡大してからは、マルチコアを実装して並列計算で高速化を図るチームも現れました。例年、発表会ではCPUを完動させたあとで「余興」として、さらに踏み込んだ挑戦が誇らしげに披露されます。自作のOSとシェルを動かす、コンパイラの最適化のためにLLVMバックエンドを作成する、などが近年の例です。そこまで熱くなれるのがCPU実験です。

(2009年3月 谷田直輝・2020年3月 山田允更新)

マルチコアで並列処理を実現するため、コンパイラの開発はイバらの道に……。仮想メモリやマルチタスクのOSを動かすためには、CPUにさらにMMU(メモリ管理機構)、割り込み機構、特権管理機構を作り込まなければならない。

研究ってどんなものだろう?自分で選んだ課題に取り組む

演習III 研究室めぐりから 卒論への道

情報科学にはいろいろな分野があり、それぞれ広範な隣接分野へとつながっています。多くの人がそのどの分野に進むかを迷い、興味を持っている分野の感触を知りたいと思っているでしょう。ならば、いくつか実際に体験してみようという制度があります。

演習IIIという独自のシステム

4年生は、Aセメスターにはいると研究室に配属され、自分で選んだテーマを研究し、得られた知見を卒業論文にまとめます。とはいっても、どの分野を選ぶか、どの研究室が合っているかは多くの人が迷い、悩むところでしょう。そして、これから進むことになる研究室では、いったいどんな生活が待っているのでしょうか?

情報科学科では、情報科学演習III(以降では演習III)という独自の制度があり、研究室に配属される前に、3つの研究室を1カ月ずつ訪問し、それだけで課題に取り組みます。この過程を通して、情報科学の異なる分野を体験し、卒論や大学院に向けて自分に合った分野を見つけることができます。

4年Sセメスター:

4月になると、分野・研究室を紹介するガイダンスが開かれ、それを参考にして希望する研究室を6つまで提出します。4月上旬には、訪問先の3つの研究室が決まり、4月中旬から1カ月間ずつ仮配属されます。

研究室で取り組む課題は、それまでの授業のように一方的に出題されるわけではありません。用意されている選択肢のな

がら自分でテーマを選び、1カ月をかけて取り組みます。

4年Aセメスター:
9月になると、配属先の研究室が決定し、早ければ9月中から研究室のミーティングに参加するようになります。研究室では、担当教員の指導を受けながら、1つのテーマを追求して、卒業論文を書き上げます。論文のテーマの決定から、実装、実験、論文執筆までは4カ月。論文は原則的に英語で執筆します。苦労が実って、国際学会での発表につなげる人もいます。この時期は皆、かなり忙しくなります。

「研究」の世界を覗く

演習IIIは、ほとんどの学生にとって、はじめての研究の体験になります。3年生までの講義や演習では、情報科学の基礎を学びますが、演習IIIのテーマは自分で選ぶものの、教科書のない最先端の問題に取り組みます。

課題は研究室によって異なります。基本的に最初にテーマを決めたあと、関連する論文を読み、その内容を自分なりに噛み砕いて研究室で発表し、論文で取り上げられている手法を実際に実装したり、試行

錯誤で新しい手法を実装したりします。そのなかで、「研究」の世界がぼんやりと見えるでしょう。それぞれの研究室によって、研究内容はもちろんですが、研究の進め方や研究室の雰囲気も大きく異なることもあります。

Aセメスターからは、配属先の研究室でさらに本格的な研究が始まります。指導教員や先輩の力を借りつつも、世界中の研究者と同じ土俵に立つことになるのです。

僕らの研究室めぐり

[村上]私はアルゴリズムやグラフ理論に興味があり、最初に今井研究室を訪問しました。巨大離散アルゴリズム・計算量理論・離散数学・量子計算のなかから興味のあるテーマを決め、関連のある最新の論文を選びます。

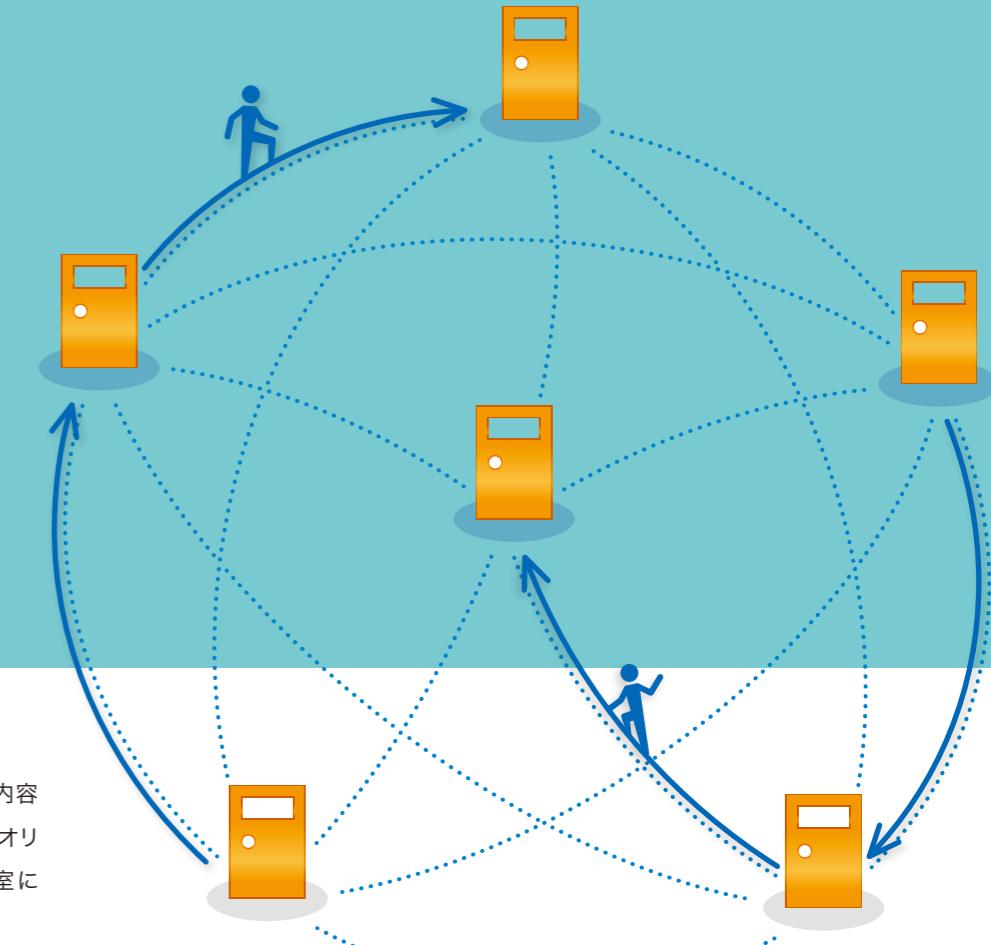
私は大規模なグラフに対するアルゴリズムについて研究することにし、KDDという国際会議でBest Paperに選ばれた論文を選びました。大規模グラフを行列として見たとき、その固有値を近似的に求める手法によって、数百万単位の頂点があっても計算が可能になります。最新の論文を読み、自分でも何か新しい発見ができるか積極的に考えるのが今井

研究室のスタイルで、論文を読んで内容をまとめるだけでなく、自分で実装やオリジナルの実験を行い、実際に研究室に入った時のイメージがわきました。

次に訪問したのは機械学習グループの研究室です。事前に機械学習のアルゴリズム・理論・応用のうちどれに興味があるのか尋ねられ、自身の興味に合ったテーマを決めます。漠然と機械学習系に興味があったものの具体的にどんな研究があるのか詳しく知らないかったです。が、興味の持てそうな内容をじっくり考えて決めることができました。最終的には、時系列系のモデルに興味がわき、いくつかの論文を読んで結果が再現できるか実験もしてみました。

最後は計算機科学グループで、私はMLIRというコンパイラの中間表現を含めたコンパイラ基盤について調べました。それほど馴染みがある分野ではなかったのですが、論文だけではなくスタイルや映像資料などが公開されていたので、それらを確認しながら進めて行きました。発表時に先生方が細かいところまで丁寧にコメントしてくださいましたので、たいへん勉強になりました。

[筒井]私はかねてからGPUを用いた並



卒業論文配属に先立ち、3つの研究室に1カ月ずつ滞在する

列計算に興味があったので、ある程度それに沿った研究ができる研究室を訪問しました。

最初に訪れた高前田研究室では、興味のあったGPUを用いた疎行列積の高速化に取り組みました。既存のGPU疎行列積アルゴリズムを、最新のGPUに搭載された行列積専用の回路を用いて動作するように改良するものです。私の書いたコードが悪かったためか、結果は従来のコードより遅くなってしまいました。

次に訪れた須田研究室でも、引き続きGPUの疎行列積の課題に取り組みました。ここでは行列積専用の回路を用いず、一般的なGPUを使用して疎行列積を高速化するアルゴリズムの改善を試みたのですが、既存の手法をCPU実装するところまで手一杯になり、GPUを使った実装まで辿り着きませんでした。

なかなか納得のいく成果は上げられなかった演習IIIですが、自分で課題を設定し、実際に手を動かして目標の達成を目指す経験は無駄ではありませんでした。実際、その後の卒業研究では、演習IIIで着目した課題とスケジュール管理の感覚が活きて、自分でも納得のいく研究成果を産むことができました。

(2022年4月 村上直輝・筒井政成)

ISerの進路

卒業までの
あんなこと
こんなこと

CHECK!



必修! 英語プレゼン集中講議

視線と姿勢、構成の組み立て方、
よりよい言い回しなど、英語での
プレゼン方法を少人数クラスで
特訓。

キョージュ面談



まずは単位を取れ。
話はそれからだ。



進路ガイダンス

研究科の「進路ガイダンス」をチェック。企業人が語る「グローバル時代に求められる人材とは」、就職活動Tips、博士課程で培った力をいかに活かすかのOBトークなど。

インターン

夏休みなどを利用して、国内・海外企業
や研究所でインターンに行く人も。



就職ガイダンス

学科・専攻の「就職ガイダンス」で、就職事情をチェック。キャリアサポート室と連携した企業の求人情報や技術説明会などもある。



CS

留学

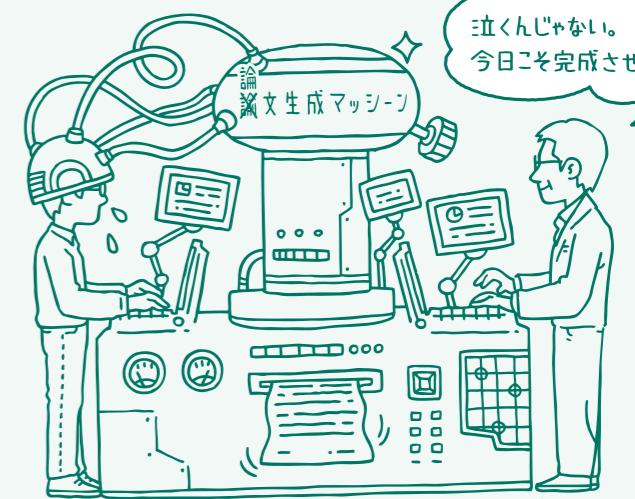
心修!

英語論文執筆講議

卒業論文は英文で提出。英語ネイティブの講師による論文執筆方法を受講。



立くんじゃない。
今日こそ完成させるんだ!!



卒業論文 締切り目前

キョージュ

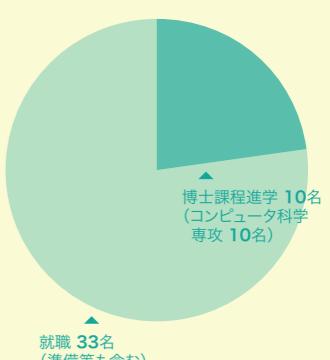
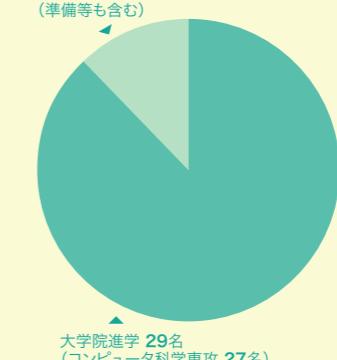
卒業後の進路

情報科学科の卒業生の多くは大学院(コンピュータ科学専攻)に進学しますが、他専攻や他大学に進学・留学したり、企業や官公庁・公共企業体などに就職したりする方もあります。

情報科学科卒業後、コンピュータ科学専攻修了後の就職先イメージは、以下のとおりです。卒業年時点の集計なので以下の数字には表れていませんが、博士課程修了後に大学に在籍したあと、企業の研究機関や海外の大学に就職する例もあります。

●2022年度情報科学科卒業生の進路
就職 4名
(準備等も含む)

●2022年度コンピュータ科学専攻
修士課程修了生の進路



情報科学科の研究室紹介

小林研究室	p.17
河原林研究室	p.18
五十嵐研究室	p.19
宮尾研究室	p.20
谷中研究室	p.21
杉山・石田研究室	p.22, p.23
佐藤研究室	p.24
横矢研究室	p.25
馬研究室	p.26
吉本研究室	p.27
品川研究室	p.28
高前田研究室	p.29

情報科学科では、コンピュータに関連する分野を扱う各研究室が協調しながら活動しています。研究室ごとに特色のある専門を持っていますが、狭い分野だけを扱うのではなく、他の領域への広がりがあります。詳しくは研究室ごとの紹介をご覧ください。

理論計算機科学

バグのない ソフトウェアを目指して

プログラム理論の深淵探究を実世界に役立てる

小林 直樹 教授 Naoki Kobayashi

ソフトウェアの高信頼化は 待ったなしの課題

飛行機や自動車から、銀行のATM、医療器具にいたるまで、いまや身の周りのあらゆるものにコンピュータが内蔵され、ソフトウェアによって制御されています。そのソフトウェアに致命的なバグ(欠陥)があったら? 考えたくないことですが、実際、ソフトウェアの欠陥による事故やトラブルは頻繁に起きており、従来のソフトウェア開発手法の限界を示しています。

そこで小林研究室では、数学的理論を駆使してソフトウェアの信頼性を高める研究を行っています。

たとえば最近では、プログラムや暗号プロトコル(ネットショッピングなどで、暗号を用いてカード番号などの機密データをやりとりするための通信方式)を検証するための理論を構築し、それに基づいた全自動検証ツールを作っています。理論を応用して、プログラムに間違いがないことや通信プロトコルの安全性を確かめられます。

プログラム理論の奥深さと 威力に魅せられて

ソフトウェアの理論は、ソフトウェアの信頼性を上げるという工学的な意味だけでなく、「学問的にみて奥が深く面白い」という点も研究の大きな動機です。

たとえば、高レベル言語で書かれたプログラムは、「ラムダ計算」というものを用いてモデル化できますが、この「ラムダ計算」は実にシンプルで奥深いものです。

「ラムダ計算」の世界には「関数」という概念しかなく、許される演算は「関数を作る」「関数を適用する」の2つのみです。にもかかわらず、これだけで実際のプログラムを記述するのに必要な概念、整数や木構造などのデータ、条件分岐、繰り返し、再帰などの制御構造まで、なんでも表せてしまいます。

このラムダ計算は、カリー・ハワード同型対応というものを通じて論理学の世界ともつながり、それがプログラム検証の土台になっています。学問的な奥深さと幅広さに加え、現代社会における重要問題の解決に貢献できること。この両面性が、ソフトウェアの基礎理論を研究する醍醐味でしょう。

高階モデル検査

現在とりわけ魅せられているのは、高階モデル検査です。モデル検査というのは、ハードウェアやソフトウェアなどを数学的にモデル化し、網羅的に検証するための技術です。発案者らが、コンピュータサイエンスのノーベル賞ともいわれている「チューリング賞」を2007年に受賞し、産業界にも徐々に採り入れられるようになりました。

高階モデル検査は、モデル検査をさらに強力にしたもので、2000年ごろから理論計算機科学者のあいだで研究されてきましたが、最近まで実際に問題を解く方法に天文的な時間がかかる非現実的な

アルゴリズムしかなく、応用もまじめに研究されていなかったのです。

ところが、我々の研究で現実的な高階モデル検査アルゴリズムが見つかり、それに基づいて世界ではじめての高階モデル検査器が現実のものになりました。それがさらに、さまざまな応用につながっています。

冒頭でふれたプログラムの全自动検証ツールは、実はこの成果に基づいています。また、高階モデル検査をデータ圧縮に応用する研究も進んでいます。文字列や木構造データを、「それを生成するプログラム」の形で表すことになると、高階モデル検査を用いて圧縮したままのデータにパターン照合や置換などの操作ができるのです。

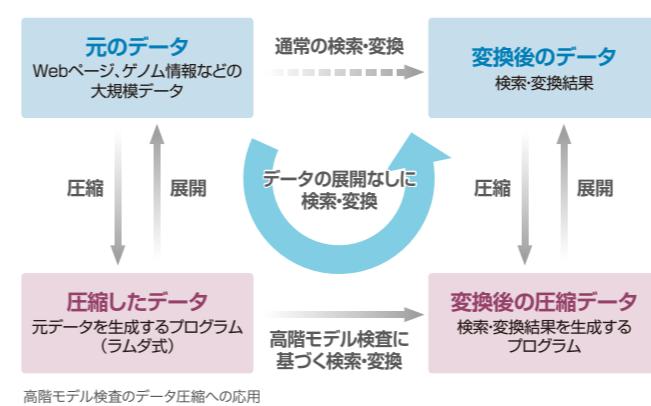
高階モデル検査は理論的に奥深く、我々の成果も「ラムダ計算」などさまざまな理論を発展・融合させて得られました。

「学問的奥深さ」と「実用性(といっても本当に実用になるのはおそらく数十年後のことですが)」の両方を兼ね備えたこのような研究テーマに出会えたことは、研究者としてたいへん幸せだと思っています。

研究テーマ

- プログラミング言語
- プログラム検証・変換
- 高階モデル検査
- ソフトウェアセキュリティ

●参考データ
小林研究室:
<http://www-kb.is.s.u-tokyo.ac.jp/>



4色問題どうしたら解けますか？

離散数学の基礎理論探求からアルゴリズムへ

河原林 健一 教授 Kawarabayashi Ken-ichi

アルゴリズムと離散数学

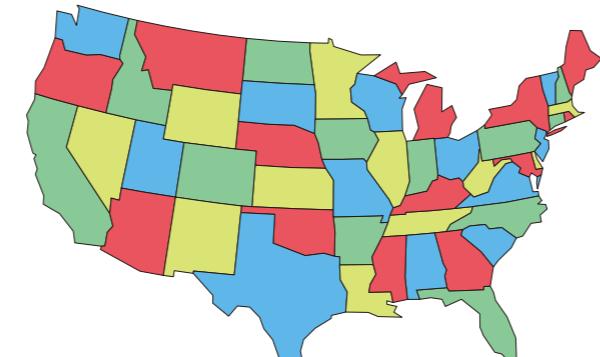
あるデータに対して、何かの計算をしたい。たとえば、SNSデータに対して最も影響力がありそうなユーザーを求めたい。この時、我々は「アルゴリズム」をプログラムとして書き、そしてデータに対して適用している。しかしデータの大きさによって、「計算時間」が莫大になることがある。この計算時間を見積もることこそ、「アルゴリズムの計算量の解析」である。アルゴリズムの計算量をなるべく小さくするための技術、そして、そのために必要な数学的道具が「離散数学」である。

本研究室では、離散数学の中でも「グラフ」を中心に研究している。グラフは、頂点とそれを結ぶ「辺」から成り立っている。SNSの例でいうと、「ユーザー」が頂点となり、「つながり」が辺になる。

4色定理

グラフに対する解析、アルゴリズムの歴史は17世紀のオイラーまでさかのぼるが、現在の「科学」としての進展は、1976年のAppel、Hakenによる「4色定理」の解決がきっかけとなっている。この4色定理とは「どんな地図も、任意の隣接2国が異なる色を

持つように、4色で塗れる」とする定理である。グラフの言葉で書くと「平面に描けるグラフは、4頂点彩色可能」とする定理となる。このルーツは1852年に地図職人であったGuthrieだといわれている。彼は、経験的にどのような地図も4色で彩色可能ということに気づき、その事実を一般的に証明できるか？という問い合わせ「4色予想」を残した。



どのような地図も、隣接する国が異なる色になるように4色で塗り分けられる

進している。これは、(競技)プログラミングコンテスト、あるいは情報オリンピックで養った手法も役立つが、線形代数、確率統計、解析、最適化などの手法を駆使する必要がある。そしてこれらを駆使できる学生と以下の目標を達成したい。

1. 計算能力を使って、4色予想(現在は定理)などの人類を悩ませる難問の解決、すなわち人類の財産である「科学」に貢献すること
2. 計算で現れる学問(離散数学、計算理論、アルゴリズム論など)を使って、他分野(物理、数学など)に本質的な貢献をすること
3. 計算能力を利用した、計算モデルに対する新しい概念を提唱すること

これらを大目標として研究を進める予定である。このような大目標を志の高い学生と共有したい。

研究テーマ

- アルゴリズム
- 離散数学
- 理論計算機科学

●参考データ
河原林研究室:
<https://kklab.nii.ac.jp/>



本研究室では、「計算能力」を最大限に発揮できるようにする離散数学研究を推進しています。

ユーザーインターフェイス

気の利くコンピュータとは？

未来のユーザーインターフェイスをデザインする

五十嵐 健夫 教授 Takeo Igarashi

タが自身の所在をGPSなどで把握していれば、人はわざわざ現在地を手で入力する必要がなくなる。気の利くコンピュータの実現には、コンピュータあるいはユーザーの置かれている状況を適切に把握し、どのような状況のときどのように動作すべきかが適切に設定されていることなどが必要である。

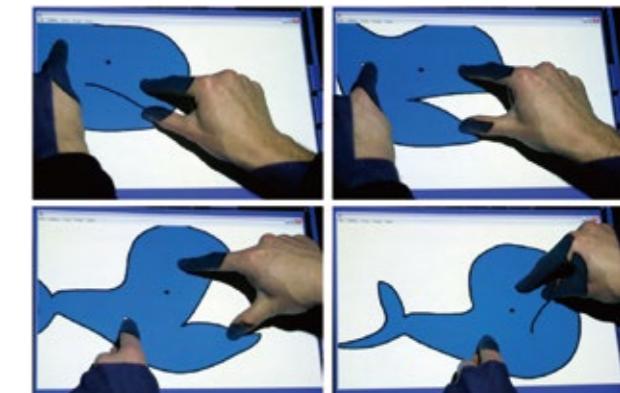


手書きスケッチによる3次元モデリング

昔のコンピュータは、何をするにも命令をいちいちキーボードから打ち込まなくてはならず、使いにくいものだったが、Windows、アイコン、メニュー、マウスを駆使したグラフィカル・ユーザーインターフェイスの普及によって、一般の人にも使えるようになった。しかしよく考えてみると、入力デバイスがキーボードからマウスに代わっただけで、人間がやりたいことをいちいち細かく指示しなければいけないことに変わりはない。このような受動的なインターフェイスは、ウェブやメールのような簡単な操作には問題がなくとも、映像の作成・編集や、他人とのビジュアルなコミュニケーションなどといった、膨大な情報をいちどに扱うような操作には適していない。また、今後家庭にはいってくると期待されるロボットのような、実世界を扱う場面にも不十分である。このような問題を解決する、未来のインターフェイスが求められている。

気の利くコンピュータ

未来のインターフェイスに必要なのは、人間がコンピュータにいちいち指示を与えるのではなく、人間の自然な動作からコンピュータが人間の必要としていることを察して手を差し延べてくれるような、「気の利く」コンピュータの実現だと、我々は考えている。バーチャルリアリティで右側を見たいときは、「右を見たい」とコマンドを打つではなく、顔を右に向ければよい。コンピュー



直接操作によるアニメーション作成手法

アイデアいろいろ

このような問題意識のもと、さまざまな新しいインターフェイスを研究している。

ひとつは、ペン入力を活用したインターフェイスのデザインである。ペン入力には、大まかな情報を手早く入力できること、また文字だけでなく絵や図も同時に入力できるという特徴があるが、既存のペン入力手法はその良さを活かしきれていない。そこで、より自由に描画しつつ高度な使い方が可能な手法を開発している。

コンピュータグラフィックス(CG)のコンテンツを手早く簡単に作成する技術も開発している。従来、CGは専門家が時間をかけて作るもので、素人が作成するのは難しかった。開発中の、手書きスケッチによる3次元モデリングや、操作の記録と再生によるアニメーション作成手法は、初心者でも簡単に3次元CGやアニメーションを作れるようにするものだ。

画像を利用したコミュニケーション支援手法、大量の情報を効率よく収集・分析・利用するための手法、また将来に向けて、家庭用ロボットを操作するためのユーザーインターフェイスも研究対象である。

ユーザーインターフェイスはまだ新しい研究分野で、解決しなければいけない

問題が多く残されている。また、個人のアイデアがすぐに世界中で使われる可能性があり、エキサイティングな分野である。より多くの人がこの分野に興味を持ってくれることを期待している。

研究テーマ

- コンテンツや設計図を簡単に作るためのインターフェイス
- 機械学習システムの構築と利用のためのインターフェイス
- ロボットや自動運転車とのインタラクション

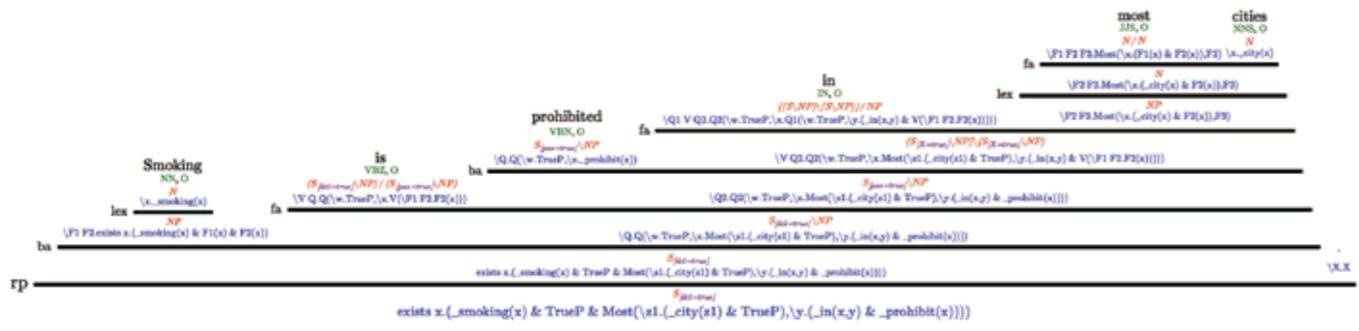
●参考データ
五十嵐研究室:
<http://www.uis.s.u-tokyo.ac.jp/>
<http://www.uis.s.u-tokyo.ac.jp/~takeo/index-j.html>



ことばと知能のしくみを解明する

ことばを理解し始めたコンピュータ

宮尾 祐介 教授 Yusuke Miyao



人間の思考の中核にある「ことば」

人間は日々いろいろな行動をしています。朝起きて顔を洗ったり、電車に揺られてしまったり、新しいアルゴリズムをプログラミングしてみたり。そのうち、どれくらいに「ことば」(自然言語)が関係しているでしょうか。

顔を洗うのはことばと関係ないように思うかもしれません。しかし、なぜ顔を洗うのでしょうか。顔を洗う理由は、おそらく誰からことばで教わったのでしょうか。寝坊したら顔を洗わずに家を出るかもしれません。その判断はどうやっているのでしょうか。プログラミングしている時はどうでしょう。頭の中で考えているとき、ことばを使っていませんか。

宮尾研究室は、人間が自然言語を理解



したり表出したりするしくみをコンピュータで再現する自然言語処理を研究しています。自然言語を理解・表出するといつても対象は広く、上記のように人間の行動ほぼすべてに関わっているといつても過言ではありません。

自然言語とコンピュータと知能

例えば、このパンフレットの原稿をコンピュータに書かせるにはどうしたらよいでしょう。1. 何を書くか考え、2. 読者が何を知っているか予測し、3. どのような順番での情報を書くか計画し、4. 最終的に文章にしていきます。これをコンピュータで再現するには、自然言語の表面的解析では不十分で、1~3のような思考プロセスやそれに必要な知識や常識も研究対象になります。

特に、ものごとを抽象的にとらえたり論理的思考をする際には言語による抽象化が不可欠で、言語は人間の知的能力の中核と考えられています。すなわち、自然言語処理とは、コンピュータを駆使し、自然言語を通して人間の知能のしくみを明らかにしようとする学問です。

具体的には、文の構造や意味を計算する構文・意味解析のような基礎研究や、質問応答、機械翻訳、対話システム、文章生成といった実社会応用を目指す研究があります。最近は、画像や数値データと自然

言語を結びつけるグラウンディングの研究もさかんで。

データの観察とモデル化

情報科学の醍醐味は、世の中のさまざまなものごとに表れる普遍的な規則性を抽象化してとらえることにあるでしょう。自然言語処理においても、英語や日本語といった個別言語を超えた「人間の言語」あるいは論理的思考といった「人間の知能」の規則性を、形式言語理論、情報論理、機械学習などを利用してモデル化する面白さがあります。

その一方で、自然言語はあくまで自然の産物であり、自分の想像はだいたい間違っていることに気づかされます。言語はこうなっているだろう、という先入観にとらわれず、実際の言語データを深く観察することが必要です。

自然言語処理の研究では、データの観察、モデル化、実験による検証というプロセスを繰り返します。これは時として失敗続きになることもありますが、それをぐぐり抜けて自然言語の新たな一端を発見する楽しさは格別です。

研究テーマ

- 自然言語の構文解析、意味解析、意味推論
- 質問応答、対話システム
- グラウンディング

●参考データ
宮尾研究室:
<https://mynlp.is.s.u-tokyo.ac.jp/ja/index>

多角的な視点から、人が「ことば」を理解する仕組みを探求する

人工知能と自然に会話できる日を目指して

谷中 瞳 准教授 Hitomi Yanaka

近くて見えぬは「ことば」

「そばにあるものほど、その実態はよくわからない」といった意味は、「遠きを知りて近きを知らず」「灯台下暗し」「傍目八目」「近くで見えぬは睫」と、実にさまざま「ことば」で表せる。そして、「ことば」はまさに、「そばにあるものほどその実態はわからない」もののなかでも最たるものである。私たちが何気なく使っている「ことば」は、コンピュータからみるとただの記号の並びにすぎず、ベクトルや論理式のようなさまざまな形式に変換することによって、「ことば」の意味を計算可能となる。

情報技術の発展とともに、「ことば」を人に代わって計算してくれる人工知能技術——自然言語処理技術——は、私たちの日常においてより身近な存在となった。例えば、レストランでメニューを見ていてわからない単語がでてきたら、とりあえずどんな食べ物なのかをウェブで検索したり、自動翻訳にかけたりすることが、日常的な所作となつた。このとき、思い通りの検索結果が出てこなかったり、おかしな翻訳結果がでてきたりしたことはないだろうか。試しに「泳げないことはない」という文を翻訳にかけてみると、「泳げない」という全く反対の意味の訳文が返ってくるかもしれない。

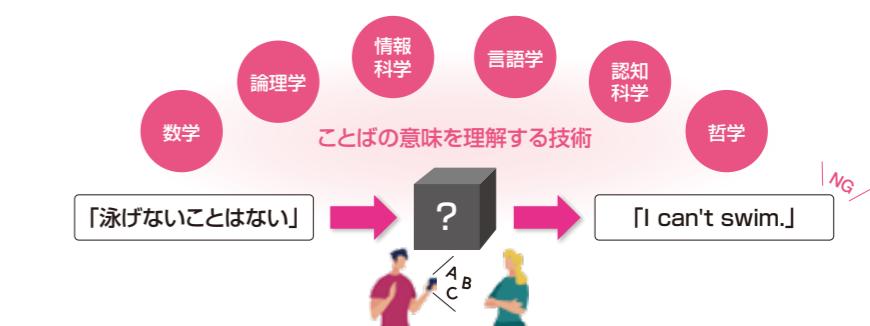
「ブラックボックス」な言語処理技術

ここで、自然言語処理技術の中身を見てみよう。最近の自然言語処理技術では、機械学習や深層学習による統計的なアプローチがよく用いられている。例えば自動翻訳技術では、日本語のデータを入力として、英語のデータを出力するように学習することで、翻訳に必要な規則を獲得していく。大規模なデータを学習すればするほど、より多くの規則を獲得でき、翻訳でき

る精度をかなり高い精度にまで高められる。しかし、このように入力と出力を直接結びつけて学習を行うアプローチでは、中間の処理過程がブラックボックス化しているため、なぜこの翻訳は正しくて、なぜこの翻訳は間違っているのかという理由を見つけることが難しい。そのため、一見するとさまざまな言語を訳すことができ賢く見える自動翻訳技術だが、翻訳の誤りを自ら省みることはできず、私たちのように「ことば」の意味を本当に理解できているのかどうかはわからない。

より人間らしい言語理解の探求

しかし、そもそも私たちはどうやって「ことば」の意味を理解しているのだろうか? この問いは、言語学や哲学、認知科学の研究に共通する、きわめて本質的な問いである。本研究室では、これらの関連分野のアプローチと情報科学や数学、論理学のアプローチとを組み合わせて、データから学習する自然言語処理技術はどこまで「ことば」の意味を理解できるようになるのか、どうすればより人間のように自然言語の意味を考え、理解する人工知能技術を実現できるのかについて、多角的な視点から探求している。多角的な視点から言語処理を考えることで、ブラックボックスを開く



研究テーマ

- 統計的言語モデルの学際的・多面的分析
- 機械学習と記号論理を融合した自然言語推論
- 人とシステムの相互作用による意味処理

●参考データ
谷中研究室:
<https://ylab.mystringly.com/>
<https://hitomiyanaka.mystringly.com/>



コンピュータはどこまで賢くなれるか？

数理によって切り拓く人工知能の未来

杉山 将 教授 *Masashi Sugiyama*

コンピュータはどれほど人間の賢さに迫るか？

少し前まで、コンピュータは、あらかじめ決められた手順どおりに情報を処理するだけの装置だった。しかし、コンピュータに自ら手順を学習させる「機械学習」とよばれる知的情報処理技術の登場によって、それまでコンピュータにできなかったことだけでなく、人がこれまで気付かなかつたことも可能になりつつある。

クイズ番組で人間のチャンピオンを打ち負かしたり、将棋でプロ棋士と互角に渡り合ったりしたりしているコンピュータの中では、まさに機械学習の技術が使われている。

検索エンジン、翻訳、通販サイトの商品推薦、CT画像からの疾患検出など、機械学習の技術は私たちの身の回りの様々な場面で活用されている。杉山研究室では、多彩な応用分野に通底する普遍的な学習原理を理論的に追求し、そして得られた汎用的な学習アルゴリズムを実世界の問題解決に役立てている。

学習するコンピュータ

機械学習は、統計的な手法によってデータの背後に潜んでいる規則性をとらえ、最

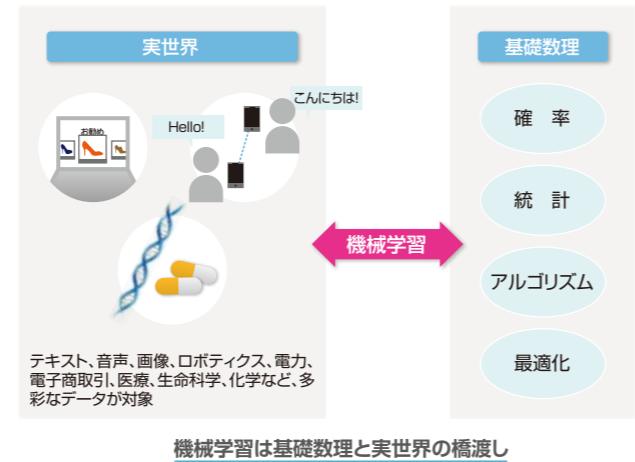
適な意志決定方法を導く。機械学習の標準的なスタイルは、人間がコンピュータに知識の一部を教え、教わっていない部分をコンピュータに推論させる「教師付き学習」というものである。うまく学習できれば、コンピュータは人が教えていなかった未知の状況にも対応できるよう

になり、人間のような柔軟で知的な情報処理が可能になる。実際、最先端の教師付き学習手法はかなり洗練されていて、一部では人間並みのレベルに達しつつある。

一方この方法は、難しい問題を解こうとするとき、人間がコンピュータに知識の一部を教える手間が大きくなってしまう。そこで、データに潜在している知識をコンピュータが自動的に抽出する「教師なし学習」への期待が高まるが、教師なし学習では、そもそもどんな知識を得たいのかがはっきりしない。そこでさらに、人手があまりかからない不完全な知識を用いる「半教師付き学習」、過去に学習した知識を再利用する「転移学習」、実世界との相互作用を通して情報を得る「強化学習」など、より柔軟な学習形態が注目を集めている。

実世界を意識しつつ抽象化するおもしろさ

ビッグデータ時代のいまは、多様なデータがインターネットやセンサーから大量に集まってくる。機械学習手法を駆使してこれらのデータを解析すると、まったく新しい知見が得られることがあり、産業や科学の発展に役立てられる。これが機械学習研究の楽しみのひとつである。



一方、それぞれのデータには特有の特徴があり、詳細にデータを解析しようとすればするほど各分野に特化した専門知識や経験が必要になる。そのため、実世界での応用を強く意識しそうると、全体を見通すことが困難になってしまいます。機械学習研究の真の醍醐味は、その抽象性にある。実世界から得られるデータを意識しつつも、その多様性に惑わされることなく学習問題を数理的に定式化することにより、そこからさまざまな分野に共通する本質的概念を見抜き、ブレイクスルーへつなげていくことが可能だ。このように、数学に根付いた確固たる基礎研究を進めつつ、実世界の難問に柔軟にアプローチしていくのが研究室の特色だ。

機械学習は、数学と実世界とを橋渡しする魅力的な研究分野である。多くの学生がこのエキサイティングなテーマに挑戦してくれるることを期待している。

研究テーマ

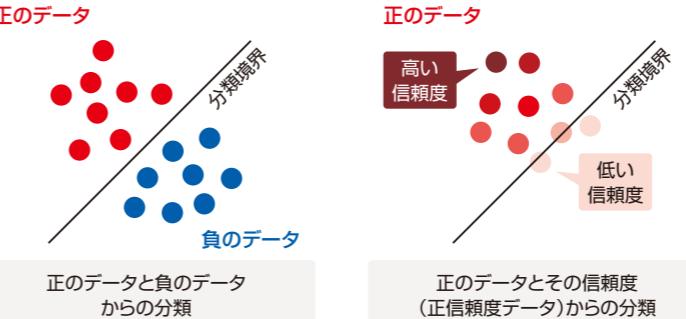
- 機械学習の基礎理論の構築
- 實用的な学習アルゴリズムの開発
- 学習技術の実世界への応用

●参考データ
杉山・石田研究室:
<http://www.ms.k.u-tokyo.ac.jp/>
<https://takashiishida.github.io/>

実用的で信頼性の高い機械学習を確立する

使いやすく、安心して使える技術へ

石田 隆 講師 *Takashi Ishida*



ものを2つのグループに分ける方法:2クラス分類(左)では正と負の両方のデータが必要。正信頼度分類(右)では、負のデータがなくても、正のデータとその信頼度がわかれれば分類境界を学習できる

開拓地が広がる機械学習

機械学習とは、コンピュータがデータからパターンや知識を自動的に学ぶ技術のことです。代表的な機械学習技術のひとつ、「分類」を例にとると、与えた写真に写っている物体が何か(それはスマートフォンなのか、キーボードなのか、コーヒーなのか?)を判断(分類)できます。与えられたデータから何かを機械的に認識・検知できるのは非常に便利なので、音声認識や物体認識、異常検知など、いろいろなところで使われています。

機械学習技術は以前からIT産業で盛んに使われてきましたが、今までとは異なる分野にも急速に需要が広がったことから、これまでとは違う新しい種類のデータを扱ったり、新しい問題に直面するケースも増え、そこに機械学習の開拓地が広がりました。

基礎技術の研究を目指す

研究室では特に、機械学習の基礎技術となるアルゴリズムを開発しています。例えば、未知のデータに対する汎化性能を高めることもそのひとつです。学習データに対して精度高く予測できても、未知データではうまくいかない、というのはよくあることです。また、データそれぞれに答えるとなる教師情報を人間が付けて学習させるこ

とも多いのですが、そのコストは高く、時間もかかります。冒頭で挙げた写真分類の例でも、機械学習を活用する前に、まずは写真をたくさん集め、一枚ずつ「これはスマートフォン」「これはキーボード」とラベルを付ける必要があります。その代わりに、もっと弱い情報、不完全な教師情報から、精度よく学習させることも研究題材です。

そのほか、センサー誤作動などによって異常データが混在していても悪影響を受けずに学習する工夫や、データを収集する環境が変化しても信頼して使えるアルゴリズムの考案なども行っています。

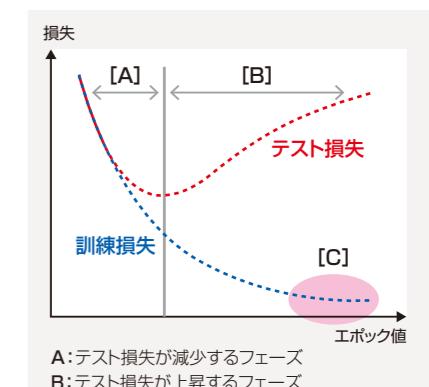
研究室の活動をまとめると、さまざまな観点からより実用的で信頼性の高い機械学習技術の確立を目指していると言えます。

何が魅力か？

機械学習の研究の面白いところは、研究の間口が広く、人によって研究スタイルが大きく異なることです。紙と鉛筆(人によってはタブレットとスタイラス)を使って数式を導出することから出発することもあれば、アルゴリズムの実装と数値実験を通して何か新しい着想を得たり突破口を見つけたりするようなケースもあります。実際に、この両者を行き来することでクリエイティブな研究に繋がることもあります。

研究は、実世界のある課題を解決した

いという具体的なモチベーションから始まっています。ところが、実際に技術を形にして論文を公開してみると、想像していなかったアプリケーションに使われて驚かされることもあるかもしれません。ある程度の汎用性を求める基礎研究ならではの面白さでしょう。ぜひ、この自由な研究スタイルと結果が広がっていく素敵な感覚を味わってください。



過学習の概念。学習を続けていくと訓練損失は下がり続けるものの、テスト損失は途中から上昇する

研究テーマ

- 弱教師学習、少数データ学習などの機械学習アルゴリズム
- 実世界における機械学習の応用

●参考データ
杉山・石田研究室:
<http://ms.k.u-tokyo.ac.jp/>
<https://takashiishida.github.io/>



機械学習を科学するための 基礎理論を築く

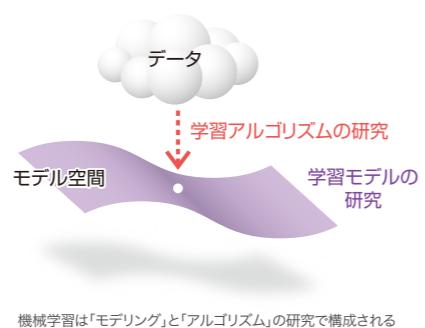
情報科学と人の知性が交差する世界

佐藤 一誠 教授 Issei Sato

「学習」とは何だろう？

人があるモノゴトを「学習した」と感じるのは、そのモノゴトについて得られた情報を「未知の問題に活用できた」時ではないでしょうか。これは計算機の学習を考える過程で行う「汎化」の基礎となる概念です。つまり「学習とは、汎化能力を向上させること」です。情報という観点では、「あるテーマに関して得られた情報を、そのテーマにおける未知の問題へ活用可能な形で抽象化すること」だといえます。

統計的機械学習は、データを数理モデルによって抽象化することで、未知の問題に対して予測を行う情報科学の技術です。汎化の観点でデータを抽象化するためには、「どのような数理モデルが良いのか」というモデリングの研究と、「データをどのように数理モデルにフィッティングするか」というアルゴリズムの研究があります(中央上図)。研究室では、このモデリングとアルゴリズムの新しい理論を構築し、実応用によって実証分析をしています。それを推し進める最大の力は、「学習」に関する深い理解と、縦横無尽に駆使する線形代数、関数解析、確率統計、最適化理論などの数学です。



機械学習は「モデリング」と「アルゴリズム」の研究で構成される

「学習」を科学する

研究室では主に、「学習」を構成するうえで重要な以下の4つの要素を理論的・実証的に分析しています。

「1 汎化と記憶」「2 摂動と不確実性」

「3 表現の学習」「4 頑健性」

これらは互いに密接に関わっており、それぞれの学習における役割を数理的に理解することはもちろん、これらの関係性を明らかにすることで「学習」またはそれに伴う「知能の創発」に関する深い理解が得られると考えています。また、そのような理解が人間の知能の理解にもつながるのではないかと考えています。

人とコンピュータが 協力しあう社会

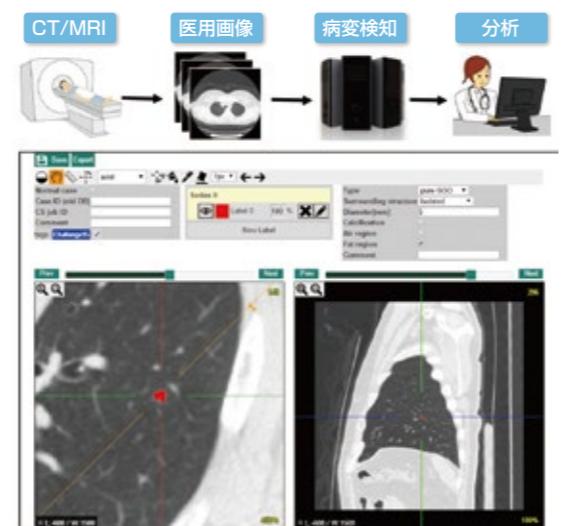
コンピュータがデータから学習することで、人の社会活動を支える新しい仕組みが生まれます。例えば、東大病院と現在共同で行っている研究では、機械学習を用いて医用画像の病変を分析し、医師の読影を支援するシステムを開発しています(右図)。

また、研究者の研究活動を支援するシステムも開発しています。一般的に研究者は、実験のデザイン、実験結果の分析、実

験設定の試行錯誤を繰り返し、研究を進めます。そこで、研究者が実験をデザインし、機械が実験と結果の分析、そして実験設定の試行錯誤を担当することで研究者を支援するという仕組みが実現可能です。

研究は点と点が つながる瞬間が楽しい

研究の世界では、それまで関連していなかった分野が結びつき、時に美しい結果を導きだすことがあります。このような瞬間を世界中の研究者とともに創造していくことは、他では経験しがたいことだと考えています。「学習」という研究を通してさまざまな分野のつながりの美しさに魅了されるのも、研究の醍醐味ではないでしょうか。



東大病院と共同開発している読影支援システム。CTやMRIでスキャンした画像を収集し、病変検出によって読影を支援する。

研究テーマ

- 柔軟な数理モデルの構築
- 高速な学習アルゴリズムの開発
- 機械学習技術の実社会への応用

参考データ

- 参考データ
佐藤研究室:
<https://www.mlis.s.u-tokyo.ac.jp/>
<https://www.mlis.s.u-tokyo.ac.jp/issei-sato-jp>

画像解析、 コンピュテーションナル イメージング

計算で見る見えない世界

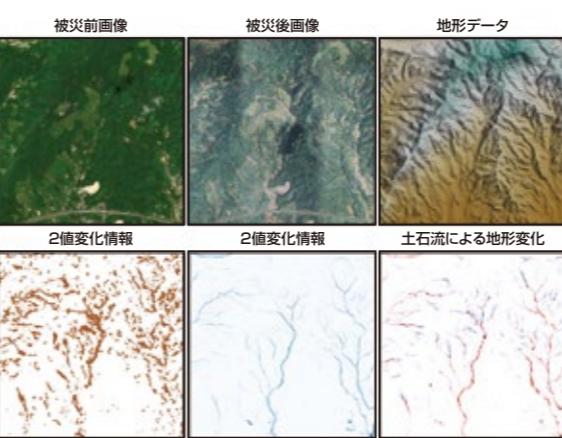
地球規模で実世界を理解する知的情報処理

横矢 直人 准教授 Naoto Yokoya

人が視覚を通して世界を認識するように、コンピュータにもカメラの画像から実世界を理解させようとするコンピュータビジョンの研究は、人工知能の一分野としてこの半世紀で大きな発展を遂げてきた。機械学習によりその技術開発は加速しており、自動運転・防犯・医療画像診断などさまざまな分野で実用化が進んでいる。人の視覚能力を超えて世界をより深く理解するために、見えない光を使った多様なイメージング技術がめざましい進歩を遂げてきた。その応用先は、ミクロからマクロまで幅広いが、カメラの性能には常に限界があり、それが画像解析のボトルネックとなっている。横矢研究室では、画像の取得と理解に関して、コンピュータによってセンシングの限界を超えることを追求している。

コンピュテーションナル イメージング

カメラの空間・時間・波長分解能やSN比などの各種性能は、トレードオフの関係にあるため、1つのカメラで得られる観測データにはハードウェア由来の不完全性が存在する。しかし、不完全な観測データから元の信号を復元してやると、解像度やノイズなどのハードウェアの限界を克服でき



シミュレーションと機械学習の融合による3次元変化認識。被災前後の画像と地形データから、2値の変化情報だけでなく、浸水深や土石流による地形変化を推定した

しか得られない場合がこの問題に該当する。3次元変化の広域計測は難しいため、機械学習のためのデータを集めることは困難だ。そこで、シミュレーションと機械学習の融合により、センシングの限界を超えた3次元変化認識に挑戦している。

情報科学で拓く地球の未来

コンピュータによる画像の取得と理解に関する研究は、実問題を解くなかで研鑽を深める分野であり、社会の役に立つ技術に直結する面白さがある。さらに、これを駆使して地球規模の問題の解決を目指すことは、何にも代えがたいやりがいがある。世界には、コンピュータとイメージングで解決すべき問題がまだ沢山あり、情報科学で地球の未来を拓く気概を持つ人材が求められている。

研究テーマ

- 画像処理や時空間データ解析
- 画像処理や時空間データの防災・環境分野への応用

●参考データ

- 参考データ
横矢研究室:
<https://naotoyokoya.com/>
https://www.ku-tokyo.ac.jp/pros/person/naoto_yokoya/naoto_yokoya.html



AI時代のソフトウェアの 安全・安心・信頼を築く

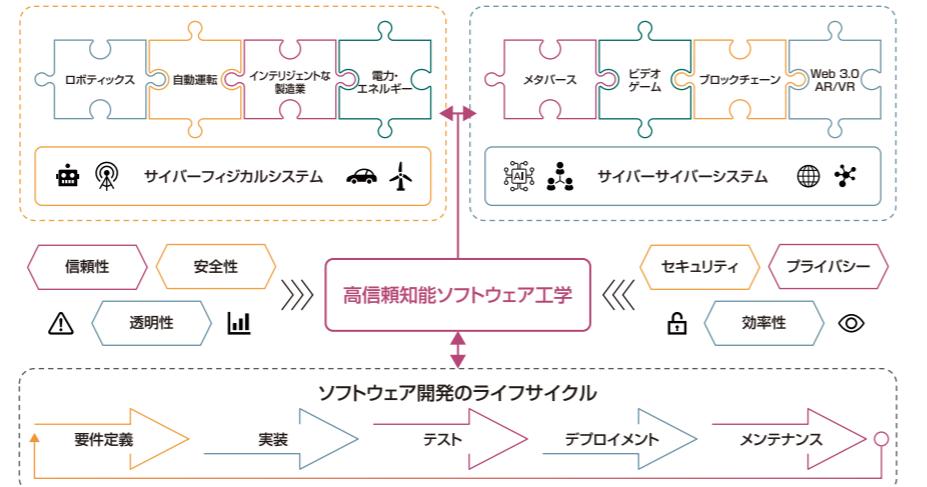
ソフトウェアの世界を探検して堅固な明日の社会基盤を築こう

馬 雷 準教授 Lei Ma

デジタル化が進んだ現代では、物理世界のモノや現象をコンピュータ上の仮想世界でシミュレートする技術が発達し、仮想世界に再現したモノや現象、あるいは物理世界と平行してできた仮想世界自体が、デジタルツイン（デジタルの双子）と呼ばれるようになっています。さらに、仮想世界に蓄積されているデータを分析し、得られる知見を物理世界に密接に還元することによって、人間の暮らしをより豊かにする未来型の社会への期待が高まっています。そのような期待を背景に、AIソフトウェアは技術革新の原動力になってきており、未来社会の基盤として不可欠です。そのため、高品質で信頼性のあるソフトウェアを構築する方法が重視されています。

データ駆動型ソフトウェアの 品質と信頼性

馬研究室では、ソフトウェア理論と工学の両面から、信頼性の高いAI応用ソフトウェアを構築するための新しい方法論や技術を研究し、一連のツール群を開発しています。特に、ソフトウェア開発ライフサイクルの主要な段階（要件定義、設計、実装、テスト、デプロイメントなど）を通して、



総合的にソフトウェア品質を確保することを提案しています。

近年の高度な知性を備えたAI応用ソフトウェアの実現には、深層学習モデルや機械学習モデルなどのデータ駆動型AIソフトウェアコンポーネントの進化が大きな貢献をしています。一方で、そのようなAI応用ソフトウェアは、従来のソフトウェアコンポーネントとAIソフトウェアコンポーネントを組み合わせた複雑な構成になっており、データ駆動型ソフトウェアに特有の課題が数多くあります。データ駆動型のソフトウェアは、プログラムの振る舞いを決める決定論理がデータから学習されますが、たとえばデータの品質と管理、学習されたモデルの品質と信頼性の分析、そして想定外のデータ混入などは、これまでになかった問題です。

AI応用ソフトウェアの品質と信頼性を高め、得られた結果を人間が理解できるものにするためには、新たな手法を編みだす必要があります。ここには研究上の大きなチャンスがあります。なお、ソフトウェアのテスト、分析、検証、モデリングなどに基づく解析などの技術には、代数、統計学、数理論理学、数学モデリング、形式論理、制御学などが大いに役立ちます。

研究テーマ

- ソフトウェア工学
 - AIシステム工学
 - サイバーフィジカルシステム
- 参考データ
馬研究室:
<https://www.malei.org>
Momentum Lab:
<http://www.momentum-lab.com/>

計算科学

科学と計算機をつなぐ

自然の不思議を解き明かすコンピュータ

吉本 芳英 準教授 Yoshihide Yoshimoto

コンピュータの発明・発展と その曲がり角

コンピュータ（電子計算機）が発明された背景には、科学技術が大量の計算を必要としているという大きな要因がありました。水や空気の流れ、電子と原子核からできている極小の世界、多数の星々から成り立つ銀河の歴史……これらの理論を構築しても、実際に計算できなければ検証も活用もできないのです。

計算機は、半導体集積回路の急速な進歩、すなわちムーアの法則に牽引されて大きく発展しました。初期のスーパーコンピュータのひとつ、CDC6600（1964年）は、1秒間に100万回程度の四則演算ができたといいます。すでに人間よりもずっと速いのですが、2015年の日本でもっとも高速なスーパーコンピュータ「京」の性能はこの100億倍です。

しかし速さの中身には違いがあります。実は、「京」は小さな計算機を約10万個も組み合わせたものなのです。つまり単体の性能は10万倍程度、それを多数組み合わせて100億倍の性能を出しているのです。これを並列化といいます。

並列化は、半導体技術の制約が顕在化して演算器単体の速度向上が困難になった2000年ごろから重要な要素となっていますが、さらに並列化の数が増し、組み合わせ方が複雑になっていることが、計算機から性能を引き出すソフトウェアづくりを困難にしています。体育祭の集団演技を思い出せば、多数の計算機を協調させてひとつ仕事を効率よく実行するのがいかに難しいか、感覚的にわかるでしょう。

そのため、計算機を活用して科学をする計算科学と計算機そのものを研究する計算機科学の関係にも変革が求められています。かつて計算科学は、年々向上していく計算機性能にただ依存することができまし

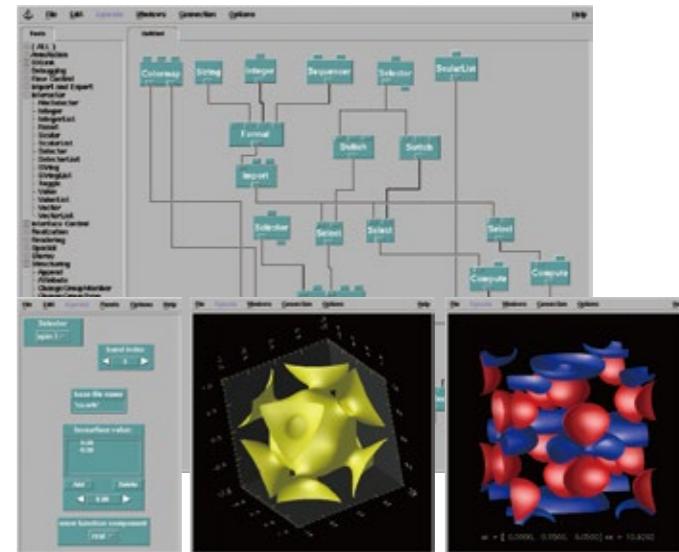
た。しかし今日、計算科学者がさらに巨大な計算を行うためには、専門分野の知識だけでなく計算機自身の理解が必要になってきています。一方計算機を設計する側にも、振り分けられる資源と目的とする計算のすり合わせが必要とされるようになりました。つまり、計算機が誕生した時と同じように、2つの分野が密接に協力する時代を迎えたのです。

再び歩み寄る計算科学と 計算機科学

吉本研究室はこのような背景で、計算機科学と計算科学の協調を、教育と研究の両面から推し進めることを目標としています。

私は計算科学をおもな専門とし、半導体、磁性体、金属、誘電体といった多様な物質の性質の大きな支配要素である電子の量子力学ができるだけ写実的に解く手法、第一原理電子状態計算を専門とし、この計算を行なうプログラムxTAPPを維持しています。一方で、国内有数の規模を持つ東京大学物性研究所のスーパーコンピュータシステムの計画運用を経験し、計算機科学への理解もあります。

研究室では、計算機科学と計算科学の境界に立ち、(1)電子状態計算をホームタウンとしつつも、計算科学の幅広い分野で個々に発展している方法論を計算機科学の観点からとらえなおし、両者をつなげること、(2)計算科学のニーズを計算機科学の



図では、Cuの第一原理電子状態計算を可視化している。下段中央がフェルミ面、右が波動関数である。下段右の赤と青の曲面はそれぞれ正と負の等高面を表しており、波動関数の節の構造が分かる。

観点から定義してより本質的な解決法を提案すること、(3)計算科学と計算機科学の相互理解を促す教育、を目指しています。

研究テーマ

- 計算科学
- 並列シミュレーションソフトウェア
(特に第一原理電子状態計算)

●参考データ

- 吉本研究室:
<http://www.cp.iis.u-tokyo.ac.jp>
xTAPP:
<http://ma.cms-initiative.jp/ja/listapps/xtapp>



コンピュータを自在に操る 特別なソフトウェア

システムソフトウェアの原理から設計、実装、評価まで

品川 高廣 教授 *Takahiro Shinagawa*

特権を持ったソフトウェア

オペレーティングシステムや仮想化ソフトウェアに代表されるシステムソフトウェアは、一般的なアプリケーションソフトウェアとコンピュータハードウェアを仲介する役割を持っています。具体的には、ハードウェアを制御する複雑な操作を抽象化して、アプリケーションに体系立てた機能として提供します。また、システム内で動作する複数のアプリケーションの間でハードウェアをうまく共有させてシステム全体の性能を引き出しつつ、アプリケーション同士を不正アクセスから保護するという一見相反する機能をバランスすることも求められます。

このような抽象化、共有、保護の機能を実現するために、システムソフトウェアには、アプリケーションがコンピュータのどの資源(CPUやメモリ、デバイスなど)にいつアクセスして良いかを決めたり、アプリケーションから直接アクセスできないハードウェアを制御したりする権限があります。

このように、システムソフトウェアはコンピュータの動作に関するすべての権限を持つ特別なソフトウェアであり、その設計だけでコンピュータの性能や機能、効率が大

きく変わります。この点が、システムソフトウェア研究の非常にチャレンジングで魅力的なところです。

システムソフトウェアの 美しさ

オペレーティングシステムをはじめとしたシステムソフトウェアを作るかというのはある意味でセンスが求められる興味深い問題です。とても美しい抽象概念を提供できると、コンピュータは使いやすくて性能が高く安全なものになりますが、行き当たりばったりに設計すると、やがて行き詰まってしまいます。

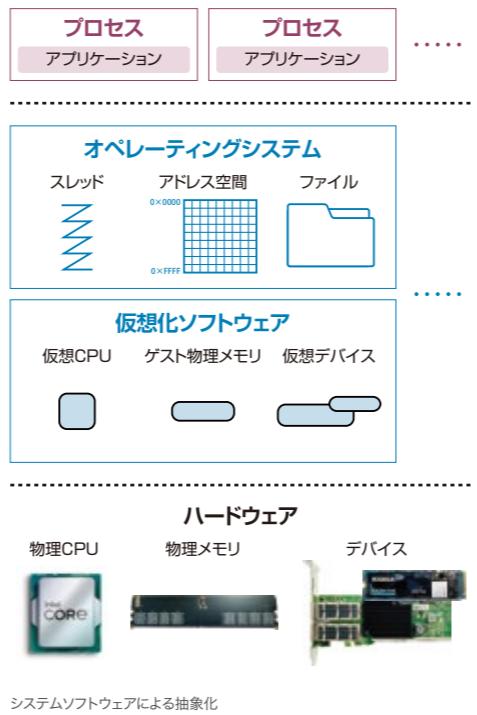
したがって、時を経ても通用する原理原則や設計思想といったものが大切になってきます。例えばOSの「プロセス」「ファイル」「ソケット」

といった抽象概念は、何十年経っても有用性を失うことがない概念として受け入れられています。

一方で、設計思想を語るだけではなく、実際のコンピュータで想定通りに動作するか評価することも大切です。コンピュータは人間が作ったものでありながら、往々にして思い通りに動作しないほど複雑になっているので、自然科学のように実験的に探っていく帰納的な手法が必要になることもあります。したがって、システムソフトウェアは理学と工学が融合した領域の学問だといえます。

最高峰のプログラミング

システムソフトウェアの研究開発には、大規模なソフトウェアの構造やハードウェアの仕様など巨大なシステムの内部構造を把握して、一貫性のある形でプログラミングする能力が求められます。これはある意味で特殊な能力であり、システムプログ



研究テーマ

- システムソフトウェア
- オペレーティングシステム
- コンピュータセキュリティ

●参考データ
品川研究室:
<https://www.os.is.s.u-tokyo.ac.jp>

コンピュータ アーキテクチャ

アーキテクチャとアルゴリズムの 協調設計でまだ速くなるコンピュータ

科学技術を牽引する、速くて使いやすいコンピュータを創る

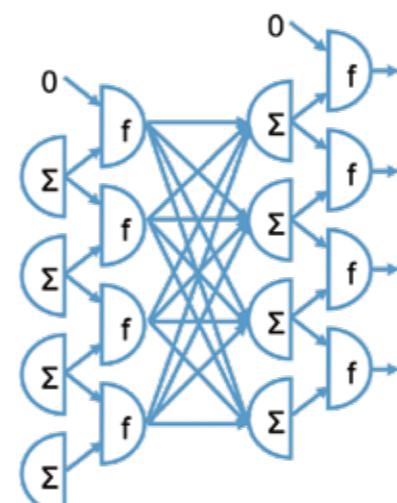
高前田 伸也 準教授 *Shinya Takamaeda*

CPUの進化に タダ乗りできる時代の終わり

コンピュータの中心的デバイスであるCPUの処理性能は年々向上し、同じ時間で扱えるデータや計算の量が増えて、ひと昔前まで原理的には可能でも速度の観点で実用的ではなかった高度な計算ができるようになっています。しかし、CPUの性能の伸びは徐々に鈍化しており、寝て待てばソフトウェアの速度が勝手に速くなる時代は終わりつつあります。そればかりか、組込み機器用の小型のコンピュータから大型のスーパーコンピュータにいたるまで、消費するエネルギー量の増大が大きな問題となっています。

汎用から利用場面に合った システムへ

そこで、特定の計算パターンに特化して高速・低消費電力に処理できるハードウェアが積極的に用いられるようになってきました。GPUはその有名な例で、広く用いられています。ひとつの命令を多数のデータ



差分二値化: 小さな回路で高い認識精度を達成する二値化ニューラルネットワーク向け活性化関数

ハードウェア性能を100%引き出すことは容易ではありません。そのため、単純に最大効率が優れたハードウェアを考えるだけではなく、自動的に性能を引き出すコンパイラ、プログラマが性能を引き出しやすいプログラミングモデル、コンピュータを扱いやすくするソフトウェアフレームワークなどの研究も進めます。

コンピュータアーキテクチャは、アイデアひとつで世界中のコンピュータシステムと、それに基づく社会システムを変容させるかもしれない挑戦的な分野です。特定の技術にとらわれず、さまざまな観点から優れたコンピュータの実現に挑戦する学生の参画を期待しています。

研究テーマ

- コンピュータアーキテクチャ
- 高位合成コンパイラ
- FPGAシステム
- アルゴリズム/ハードウェア協調設計
- 機械学習処理の高速化

●参考データ
高前田研究室:
<https://sites.google.com/view/casys-ja/>



