MWS2022 プレミーティング

研究テーマの決め方、取り組み方

NTTセキュリティ・ジャパン株式会社
研究員
碓井 利宣

2022/7/12

自己紹介：碓井利宣

- 所属：
  - NTT社会情報研究所（～入社7年目）
  - NTTセキュリティ・ジャパン株式会社（今年から出向）
- おもな経歴：
  - 学部時代から現在まで、一貫してマルウェア対策の研究に従事
    - CSS/MWS2011から論文発表
    - 「スクリプト実行環境に対する○○機能の自動付与手法」をCSSにて連載中
    - MWS Cupにも学生時代に複数回参戦
  - 2015年に修士卒で現所属に入社
  - 2018年に社会人学生として博士課程に進学、2021年に博士号を取得
- 趣味：
  - プログラム解析、CTF（セキュリティ技術の競技）、水泳、自転車
    - 趣味が高じて研究者になったタイプ

はじめに：今日のお話について

- 主に学生さん向けに、研究テーマをどう決めたか、研究にどう取り組んだかを話してほしい、とのこと

- 研究テーマの決め方、取り組み方は研究分野や人によって様々
  - 理論寄りか応用寄りかで異なる
  - MWS系かシステム系かで異なる
  - MWS系でも、研究のゴールが解析か検知かで異なる
  - どういうアプローチが好みかで異なる
  - …など

- 今日は「私が研究をする際の方法」のお話
  - 個人的な経験に基づくお話であることに注意
    - 各スライドのタイトルの先頭に「碓井の」と付けてお読みください

論文の概要（聞いてよく分からなくても大丈夫です）

NTT

- MWS2021論文「スクリプト実行環境に対する実行遅延・実行停止を回避する機能の自動付与手法」
  - 悪性なスクリプトを動的解析するシステムの研究
- **研究のモチベーション**
  - 長大なループを用いた解析妨害による実行の遅延
  - 例外による実行の停止

  } を引き起こす検体
- **課題**
  - バイナリ向けの既存技術 HASTEN[1] をスクリプトにどう適用するか
  - 特に、内部仕様が未知のスクリプトに制御フローグラフをどう構築するか
- **アプローチ**
  - まずスクリプトエンジンを解析、分岐を司るVM命令を検出
  - 分岐命令を基に制御フローグラフを構築、HASTENの手法でスキップ

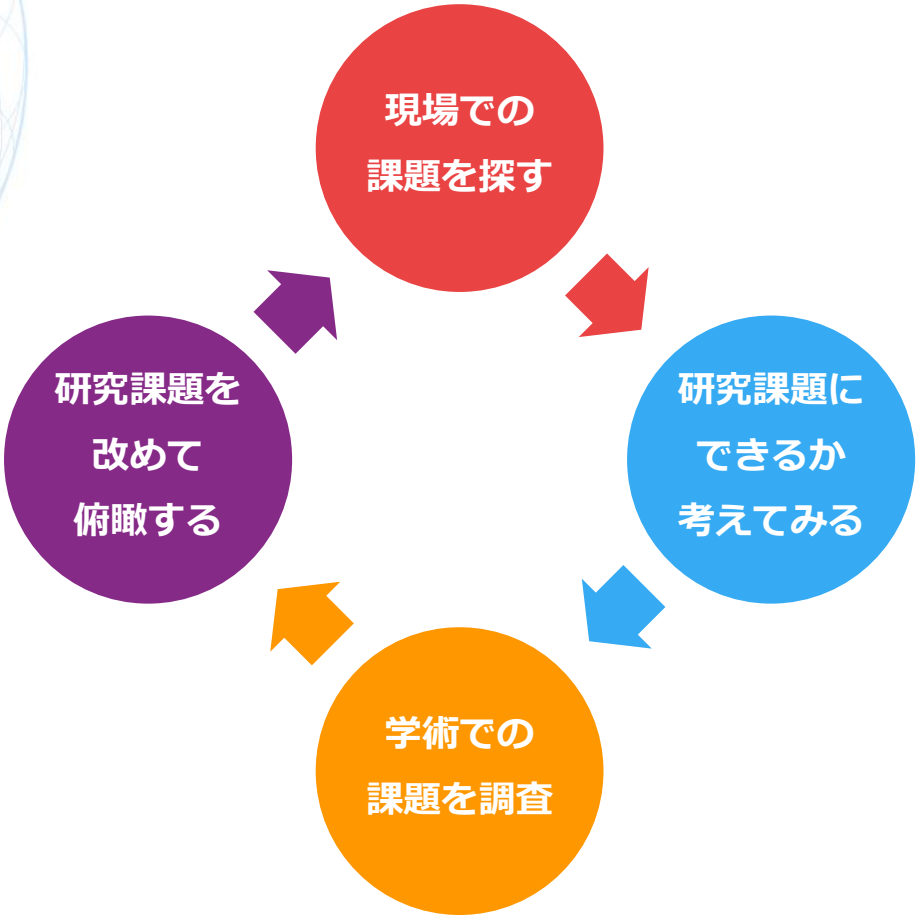
以降の話は上記の論文に基づくものです。

研究テーマ、

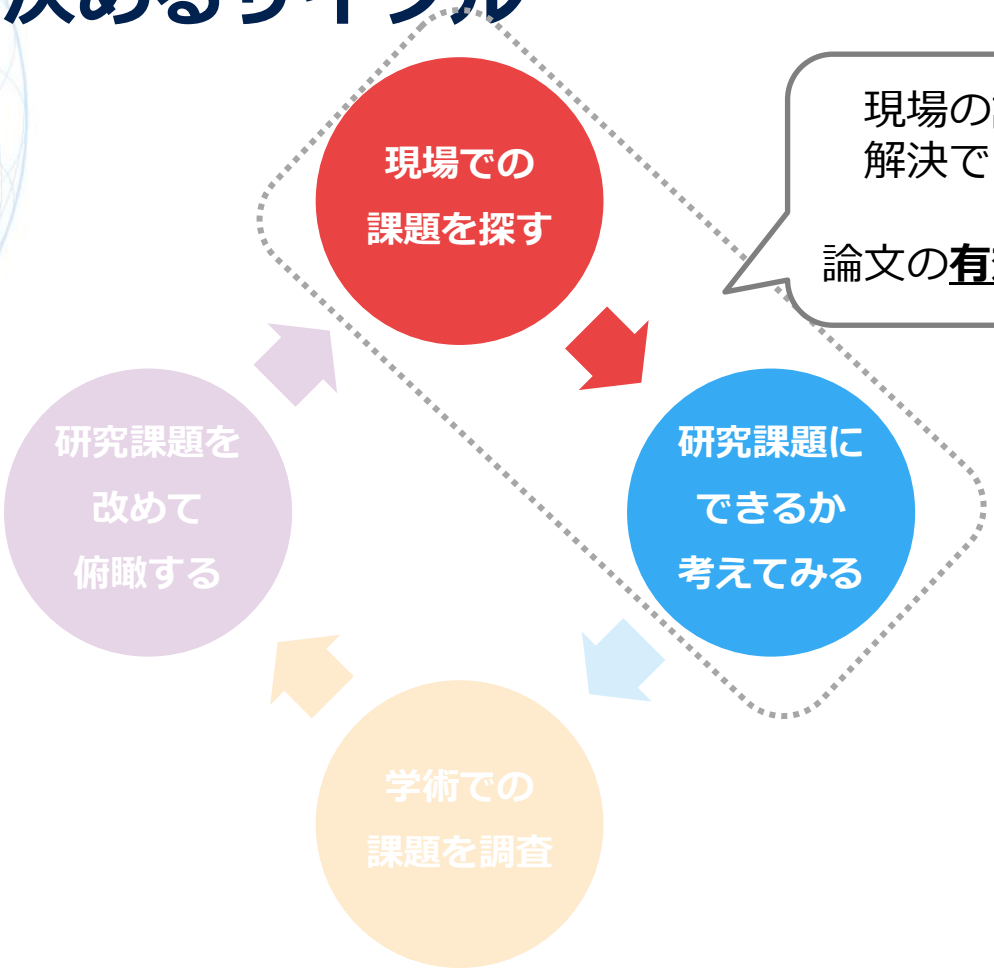
下から決めるか？

上から決めるか？

研究テーマを決めるサイクル

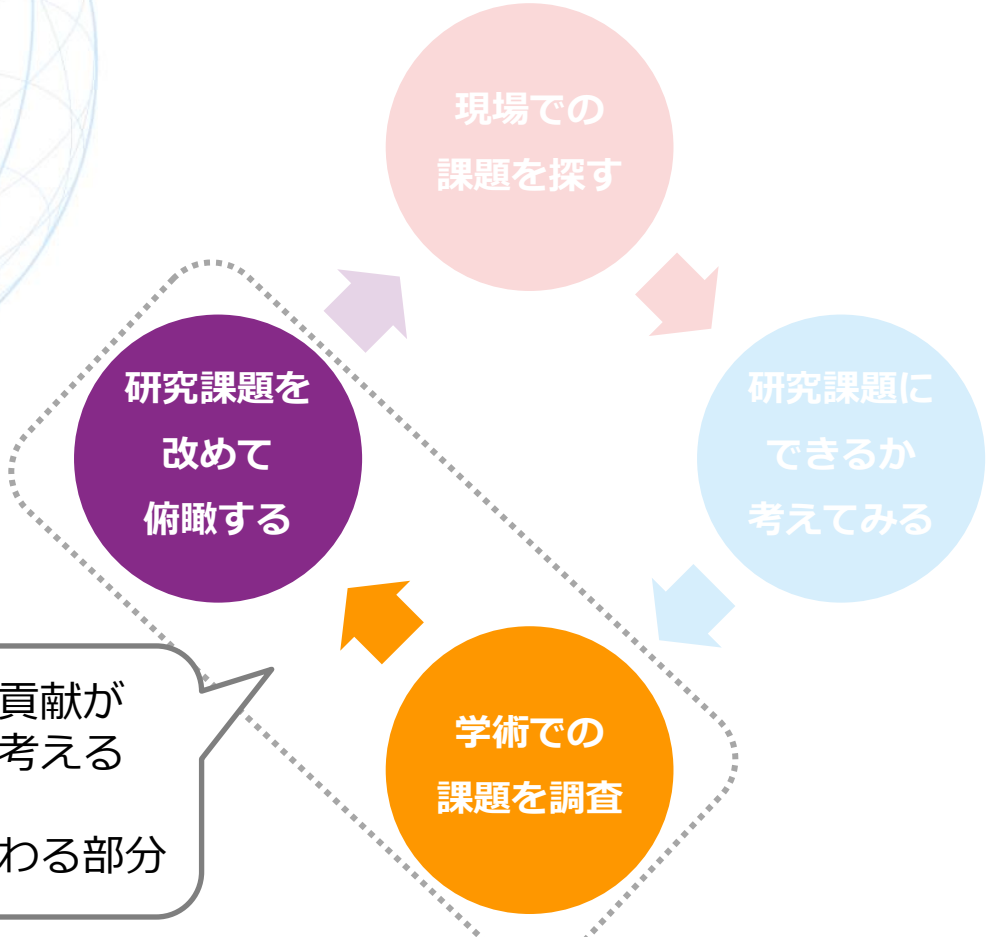


研究テーマを決めるサイクル



現場の課題を実用的に
解決できるかを考える
↓
論文の**有効性**に関わる部分

研究テーマを決めるサイクル



同時に学術的な貢献が
実現できるかを考える
↓
論文の**新規性**に関わる部分

研究テーマを決めるサイクル

現場での

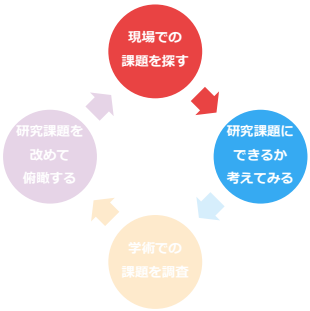
実用からの要請に基づく**トップダウン**と
学術の課題に基づく**ボトムアップ**を両立

↓

論文に重要な**有効性**と**新規性**を確保する

学術での
課題を調査

研究テーマの決め方 (1/2)



① 現場での課題を探す（現場百遍！）

- マルウェアを実際に自分の手で解析する
  - 碓井@MWS2021「動的解析を妨害する検体があるのに気付いた」
- 既存のツールを調査したり試したりしてみる
  - 「調べた範囲では、対処できる良さそうなツールはない」
- 最前線の現場にヒアリングする
  - 「OSSのスクリプトエンジンを独自に改造して、手動解析で凌いでいる」
  - 「悪性スクリプトは多様なので、言語に汎用的なツールがあると嬉しい」

② 研究課題にできるか、**まずは実用的視点から**考えてみる

- 既存のツールや技術の組み合わせで解決できる安易な課題でないか考える
  - 自分が現場の解析者だとして、その課題を解決するツールが現れたら嬉しいか？
- 切り口や解決方法を考えながら、いくつか候補のアイデアを挙げる

研究テーマの決め方 (2/2)

③ 学術での課題を調査

- 関連分野の主要な論文を読む（30～50本）
  - 「その分野の研究者なら当然読んでいるであろう論文」は全て読む
    （サーベイ論文やRelated Workの章をヒントに）
- 研究分野の流れをグラフにまとめる（次ページ）
  - 分野の開祖と言える論文からどう進化してきたか

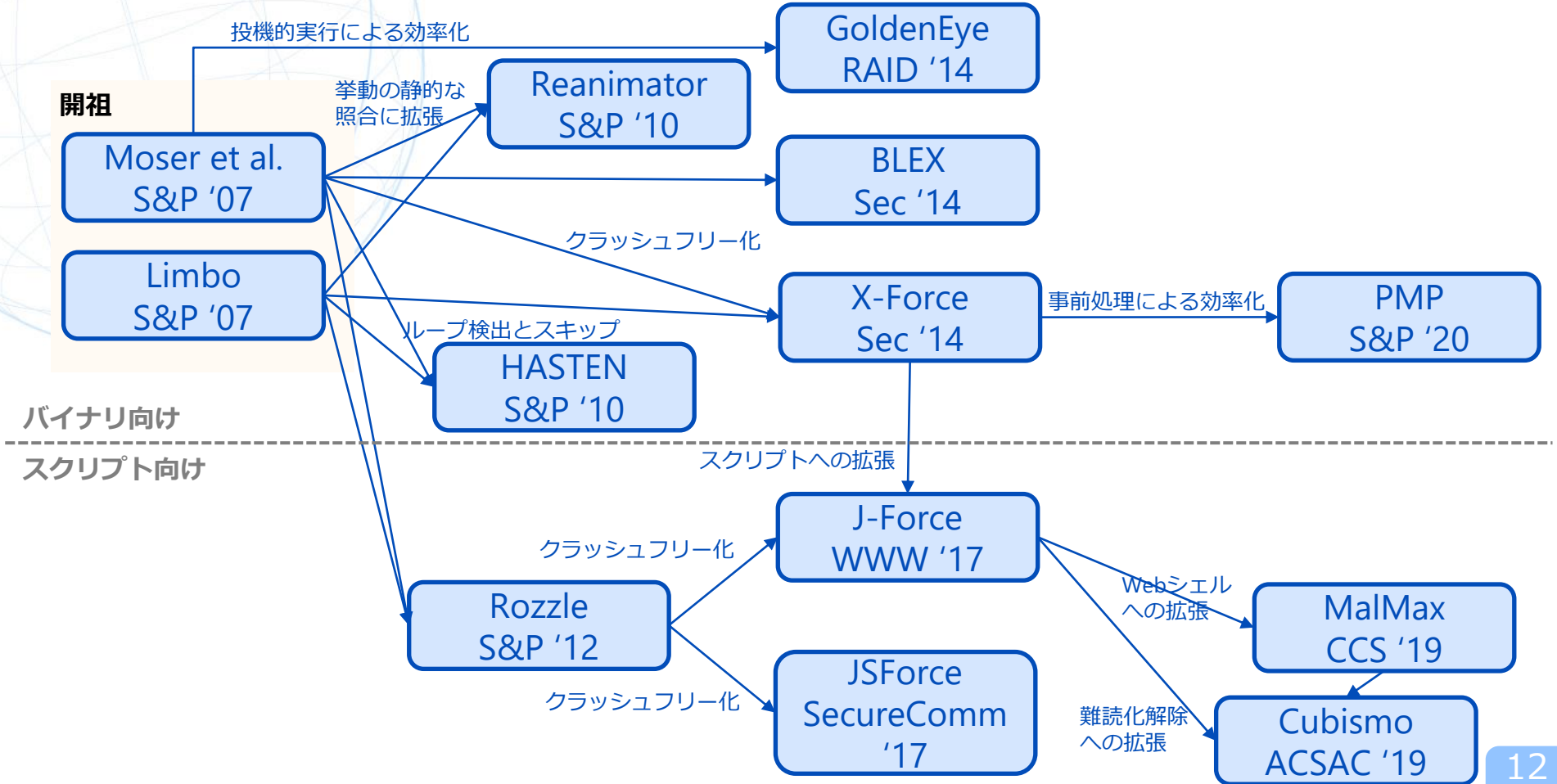
④ 研究課題を**学術的視点から**改めて俯瞰する

ここでよく死ぬ

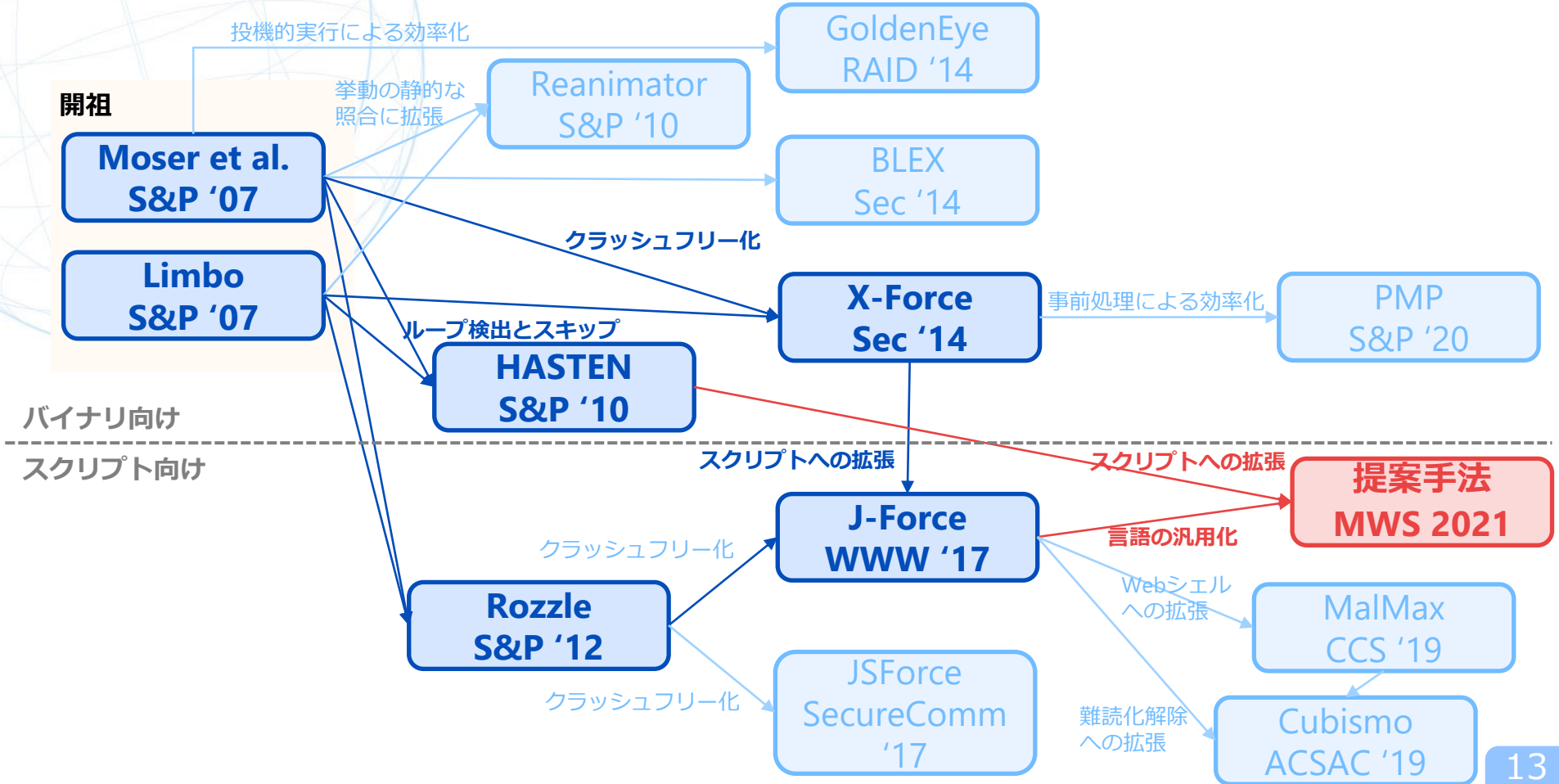
- ②で挙げたアイデアに、学術的な新規性や面白さがあるか考える
  - または、どうしたら新規性や面白さを見出せそうかを考える
- 先ほどのグラフの中で自分の研究はどこにマッピングできるかを考える

②と④のチェックを通過したら、研究テーマに決定
⇒ 基になった①のマルウェア検体は Motivating Example に据える

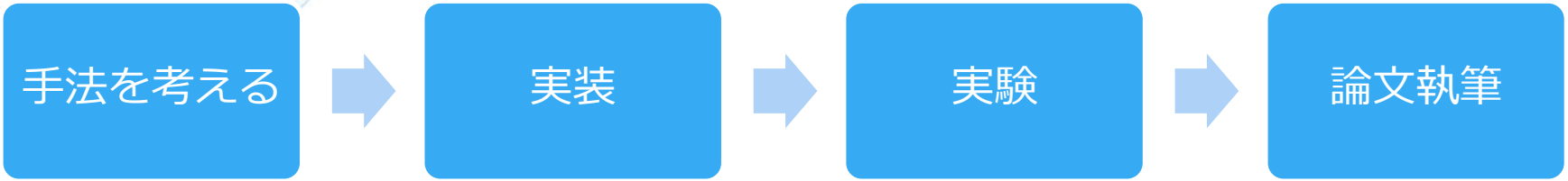
グラフ（一部抜粋）


開祖
投機的実行による効率化
GoldenEye
RAID '14
挙動の静的な
照合に拡張
Reanimator
S&P '10
Moser et al.
S&P '07
BLEX
Sec '14
クラッシュフリー化
Limbo
S&P '07
X-Force
Sec '14
事前処理による効率化
PMP
S&P '20
ループ検出とスキップ
HASTEN
S&P '10
バイナリ向け
スクリプト向け
スクリプトへの拡張
J-Force
WWW '17
クラッシュフリー化
Webシェル
への拡張
MalMax
CCS '19
Rozzle
S&P '12
JSForce
SecureComm
'17
クラッシュフリー化
難読化解除
への拡張
Cubismo
ACSAC '19

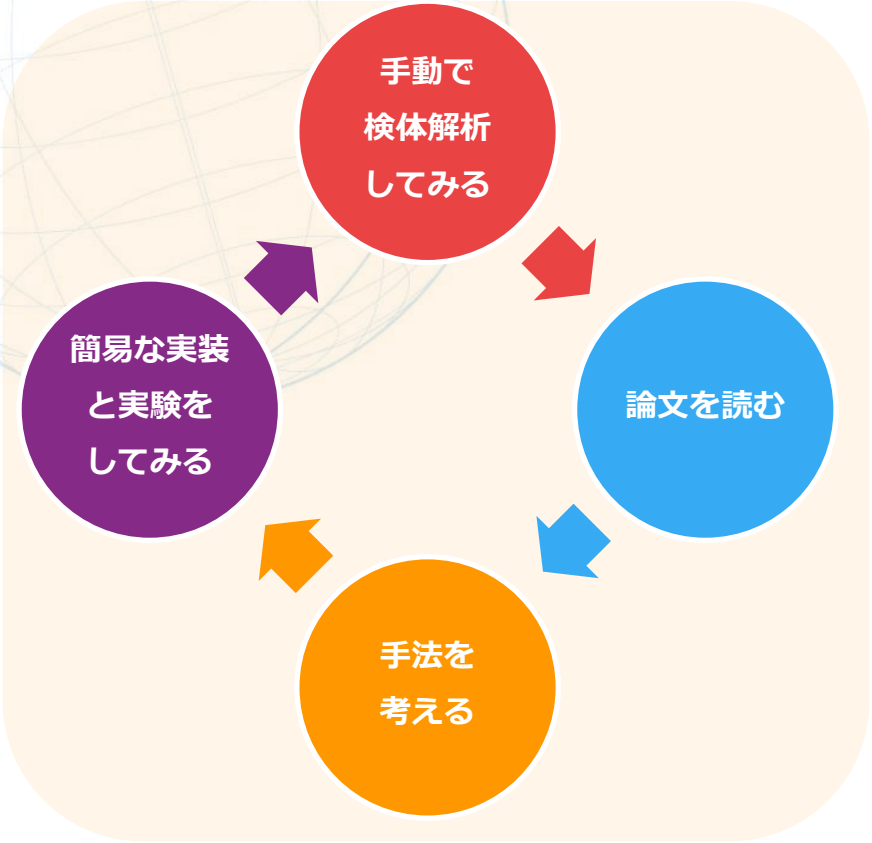
自分の研究テーマのマッピング

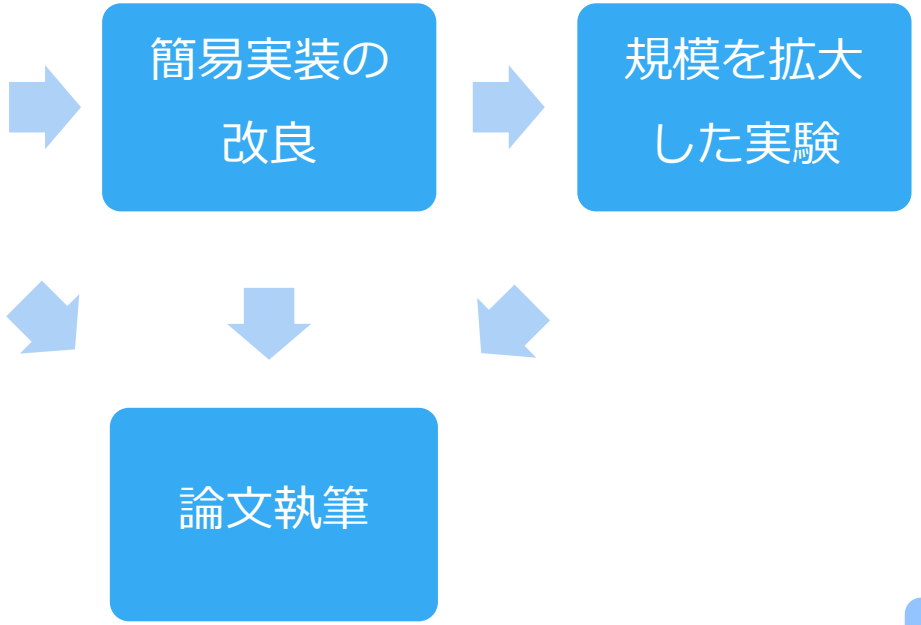


研究テーマの取り組み方（昔）

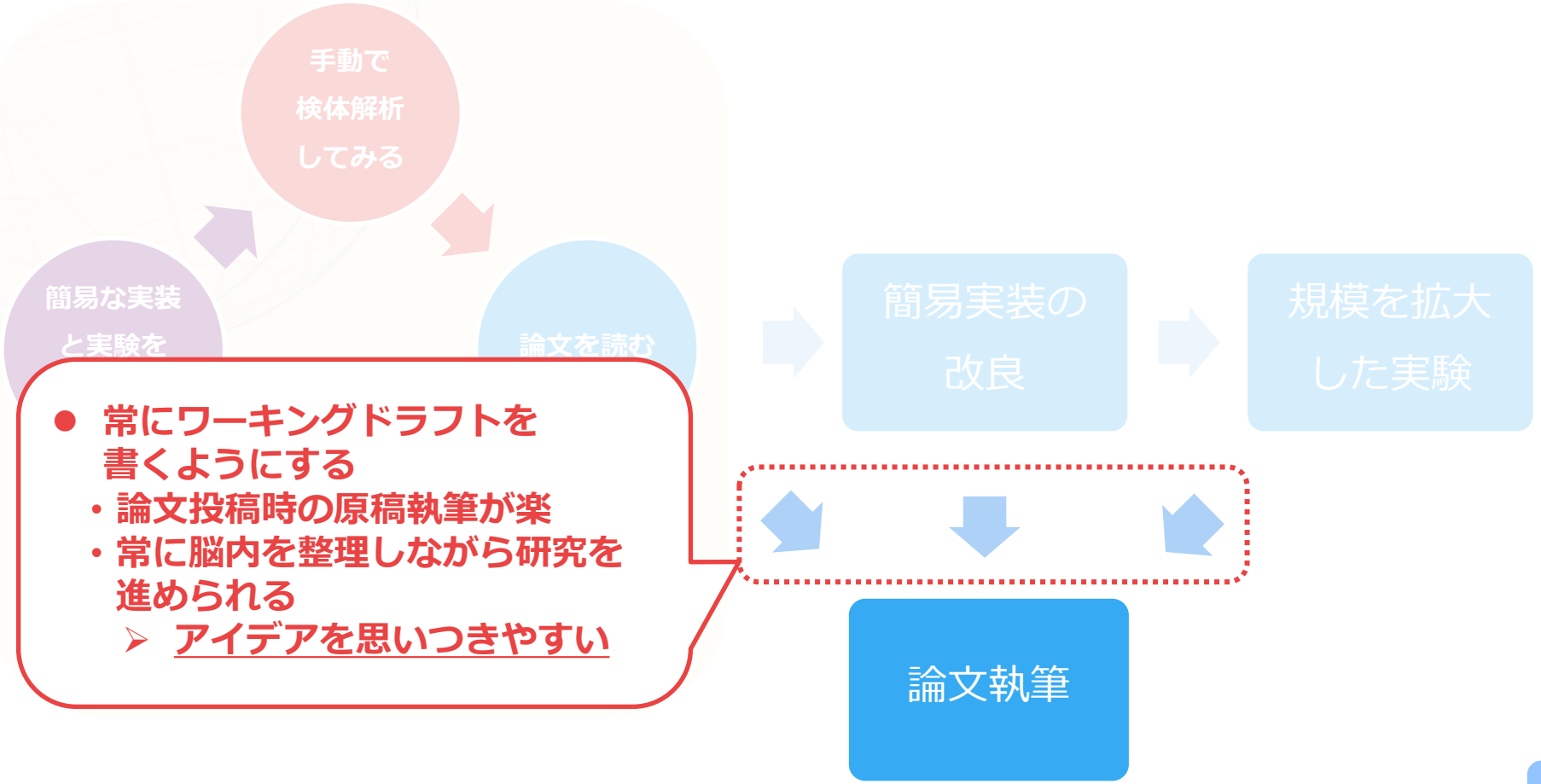


研究テーマの取り組み方（現在）



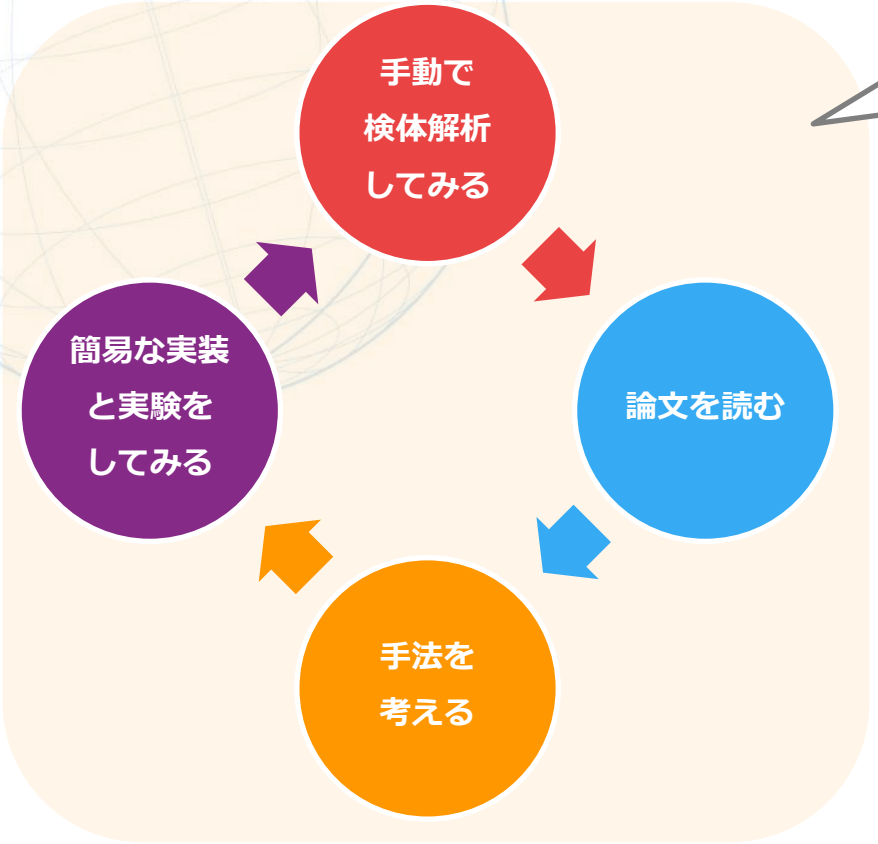


研究テーマの取り組み方



- 常にワーキングドラフトを書くようにする
  - 論文投稿時の原稿執筆が楽
  - 常に脳内を整理しながら研究を進められる
    - ➢ アイデアを思いつきやすい

研究テーマの取り組み方



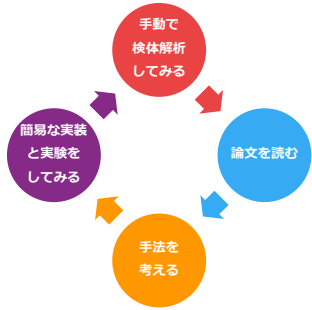
Motivating Exampleが
想定通りに解析できるようになるまで回す

簡易実装の
改良

規模を拡大
した実験

論文執筆

手法の検討 (1/2)



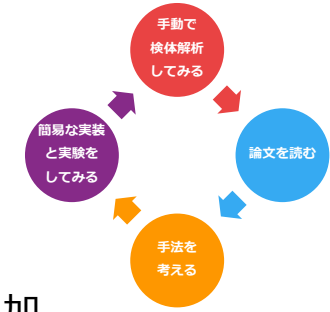
① 手動で検体解析してみる（現場百遍！）

- Motivating Exampleを解析する
  - ②や③で手法を考えていく参考にする
  - ④で簡易実験がうまくいかなかった場合の原因分析をする

② 論文を読む

- 研究テーマ決めの時と違い、手法に関するメタな知見を得る読み方をする
  - 提案手法の背景にある問題解決のパターンを考える
    - 事前に可能な限り計算しておいたり、計算済みのものをメモ化したりして高速化
    - 未知のデータ列と既知のデータ列を比較して知見を得る
    - 出力が予測可能なテスト入力を用いてプログラムの振る舞いを事前学習　…など
- 有用な問題解決のパターンは分野によらない場合もある
  - たとえば、Webセキュリティの論文がマルウェア解析に生きることもある
  - 日頃から色々な論文をメタに読むのが大切かも？

手法の検討 (2/2)



③ 手法を考える

- ②で得た知見を組み合わせたり、独自に考えたりして考案
  - 可能な限りシンプルな手法にできるよう気を付ける
  - MWS2021論文=HASTEN[1]+Virtuoso[2]+Carmony et al.[3]+独自手法
    - HASTENをベースに、他2つの問題解決パターンを生かした独自手法を追加

④ 簡易な実装と実験をしてみる

- 手法のテストのようなもの
  - 考案した手法でMotivating Exampleを解析できなかったら、再検討が必要
- 経験上、手法が一発で想定通りに解析できることはまずない
  - 研究テーマに固有の問題が発生したら、手法の改良のチャンス
  - MWS2021論文も、HASTENにはない問題に悩まされたが、適宜手法の改良で対応

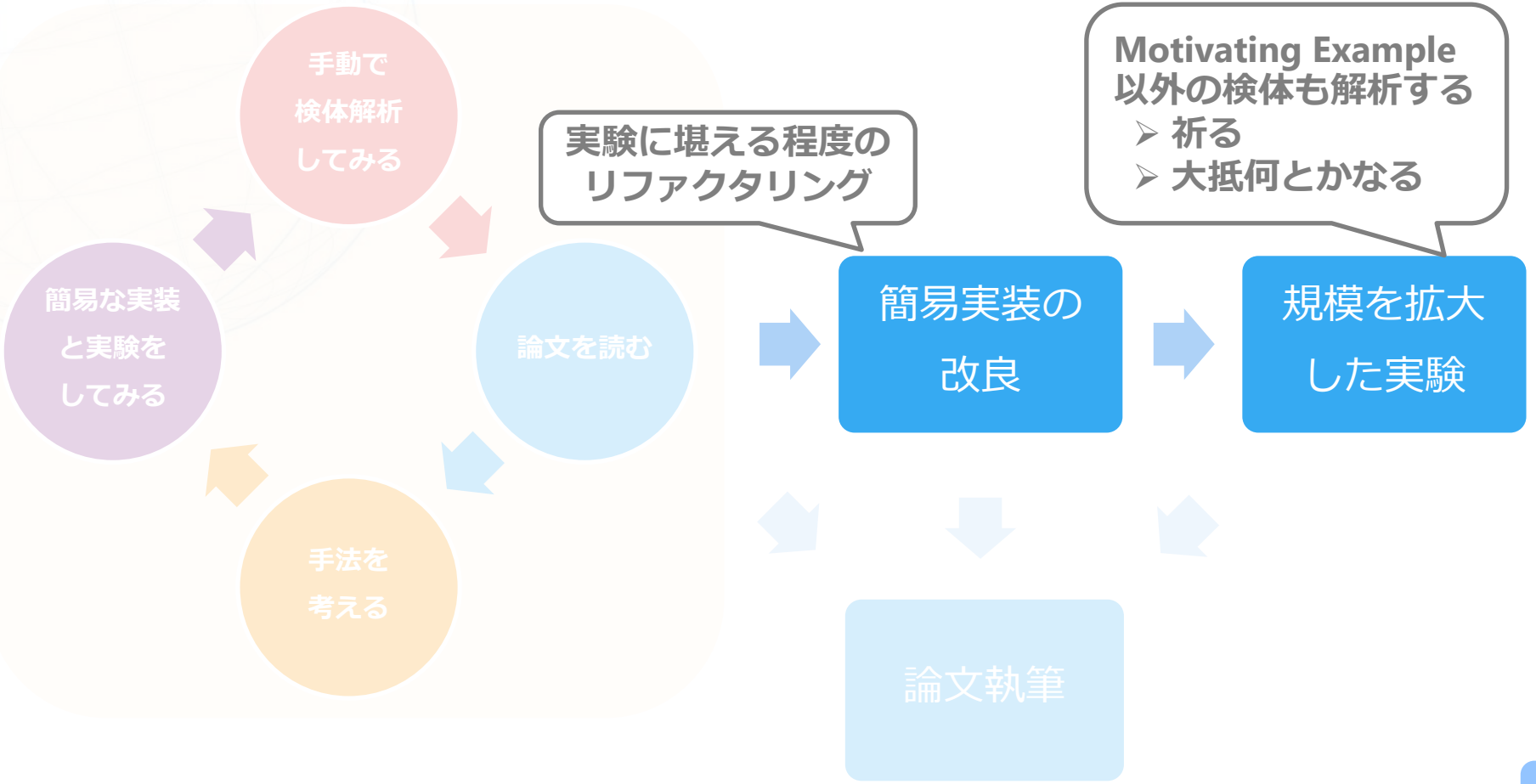
④のテストを通過したら、ひとまず論文にはなる
⇒ あとは実験の規模を拡張していく

手法の検討時に考えること

"Some seek complex solutions to simple problems; it is better to find simple solutions to complex problems." - Soramichi Akiyama[4]

「シンプルな問題に複雑な解決方法を探す人々がいるが、複雑な問題にシンプルな解決方法を探す方が良い」穐山空道

研究テーマの取り組み方



CSS/MWSコミュニティを活用しよう！

- MWSデータセットを研究に生かす
  - マルウェアの能動的な収集が容易でない昨今、データセットは貴重な情報源
- MWS Cupをマルウェア解析の技術研鑽に生かす
  - 自身の解析技術が高い方が、研究に実用性を持たせやすい（現場百遍！）
    - 最新の検体を解析できないと、研究のチャンスを逃すことも…
- 頑張って研究して査読結果をもらい※、さらなる研究に生かす
  - 個人的に、MWSの査読は結構的確だと思う
    - MWSで評価された論文は難関国際会議にもしばしば採録される
    - 難関国際会議の査読者もしばしば同じようなコメントをしてくる
      - MWSの査読で好評だった点は、大抵は国際会議の査読でも好評
      - MWSの査読で指摘された点は、直しきれないと国際会議の査読でも指摘される

※MWSトラックにて、選奨の際に一定以上の評価をとると後日届く

参考文献

[1] Kolbitsch et al., The Power of Procrastination: Detection and Mitigation of Execution-Stalling Malicious Code, CCS ‘11

[2] Dolan-Gavitt et al., Virtuoso: Narrowing the Semantic Gap in Virtual Machine Introspection, SP ‘11

[3] Carmony et al., Extract Me If You Can: Abusing PDF Parsers in Malware Detectors, NDSS ‘15

[4] Montanez et al., Inertial Hidden Markov Models: Modeling Change in Multivariate Time Series, AAAI ‘15