IMES DISCUSSION PAPER SERIES

共通鍵暗号型の検索可能暗号の処理性能について

芦原聡介(あしはらそうすけ)・清藤武暢(せいとうたけのぶ)

Discussion Paper No. 2017-J-7

IMES

INSTITUTE FOR MONETARY AND ECONOMIC STUDIES

BANK OF JAPAN

日本銀行金融研究所

〒103-8660 東京都中央区日本橋本石町 2-1-1

IMES Discussion Paper Series 2017-J-7
2017 年 3 月

共通鍵暗号型の検索可能暗号の処理性能について

芦原聡介(あしはらそうすけ)*・清藤武暢(せいとうたけのぶ)**

要　　旨

近年、金融機関においても、情報システムを効率的に構築・運用するため、クラウド・サービスを提供する外部事業者へ業務をアウトソース化する動きが進展している。金融機関がクラウド・サービスを利用する際の主な懸念は、サービスを提供する外部事業者から、機密性の高いデータが漏えいするリスクである。こうしたリスクの軽減に資する技術として「高機能暗号」が注目されている。検索可能暗号は、代表的な高機能暗号の 1 つであり、データを暗号化したままで検索を可能にする技術である。同技術は、クラウド・サービス上でデータの機密性を確保しつつ、大量のデータの中から必要なデータを効率よく取得することを可能にするものとして期待されている。本稿では、共通鍵暗号型の検索可能暗号を取り上げ、その研究動向を紹介する。また、処理性能にかかる評価尺度を示し、最近提案されている主要な実現方式の比較を行う。

キーワード：共通鍵暗号、クラウド・サービス、検索可能暗号、高機能暗号、処理性能
JEL classification: L86、L96、Z00

*　日本銀行金融研究所（E-mail: sousuke.ashihara@boj.or.jp）
**　日本銀行金融研究所（E-mail: takenobu.seitou@boj.or.jp）

本稿の作成に当たっては、長崎県立大学の松崎なつめ教授から有益なコメントを頂いた。ここに記して感謝したい。ただし、本稿に示されている意見は、筆者たち個人に属し、日本銀行の公式見解を示すものではない。また、ありうべき誤りはすべて筆者たち個人に属する。

目 次

１．はじめに

近年、幅広い分野で、情報システムを効率的に構築・運用する方法の 1 つとして、クラウド・サービス[1]を活用して業務をアウトソースする動きが広がっている。そうした中、金融機関においても、クラウド・サービスへの業務のアウトソース化が進展してきている。アウトソース化の主なメリットとしては、システム開発・運用コストの削減やシステム導入の迅速化、およびサーバやストレージ等のシステム・インフラの拡張性の向上等が挙げられる。金融情報システムセンターによるアンケート調査[2]によれば、アンケートに回答した金融機関等の約 4 割が、クラウド・サービスを利用して、営業支援システム、社内の情報共有システム、電子メールシステム等を構築・運用しており、その利用率はさらに増加する傾向にある（金融情報システムセンター［2016］）[3]。

もっとも、アンケート回答先の約 8 割がクラウド・サービスを利用する際の懸念として、サービスを提供する外部事業者から機密性の高いデータが流出するリスクを挙げている（金融情報システムセンター［2016］）。こうしたリスクは、外部事業者に対する立入監査やモニタリング態勢を整備するなど、金融機関が外部事業者のセキュリティ管理の状況を適切に把握することによって制御することができると考えられている（金融情報システムセンター［2015］）。しかし、そうした対応には相当なコストが発生し、アウトソース化の主なメリットであるコスト削減が実現しない場合があると考えられる。金融機関がコスト削減等のアウトソース化のメリットを享受するためには、クラウド・サービスに預託するデータの安全性にかかるリスクをいかに効率的に制御するかが重要な課題であるといえる。

機密性の高い大量のデータを外部業者に預託する場合、その前にデータを暗号化することによって機密性を確保することが考えられる。しかし、単純に暗号化だけを行って預託すると、預託された（暗号化済みの）データから必要な情報を検索・抽出することができなくなる。したがって、データをいったん復号してから検索を行うということとなる。しかしこの場合も、外部事業者による復号を回避するために、データ全体をいったん外部事業者から取戻したうえ

---

[1] クラウド・サービスの利用形態には、複数の組織等がストレージや CPU 等の計算資源を共有して利用するパブリック・クラウドや、独立した 1 つの計算資源を個々の組織が占有して利用するプライベート・クラウドのほか、複数の組織等が共同体を形成して 1 つの計算資源を利用するコミュニティ・クラウドが含まれる。

[2] 当該アンケートでは、従来から存在する、勘定系システム等の共同利用や共同センター、および共同利用型のインターネット・バンキングは、クラウド・サービスには含まれない。

[3] これらに加えて、金融機関におけるクラウド・サービスの利用事例としては、業務で作成したデータの共有、クラウド・サービスで提供される計算資源によるデータ分析、顧客向けの情報提供ポータルサイトの構築等が知られている（日立製作所［2013, 2014］、Box [2016]、アマゾン・ウェブサービス［2016］）。

で復号する必要がある。そのため、利便性や効率性の観点からクラウド・サービスを活用するメリットが低下してしまう。

こうした問題を解決する技術として、「検索可能暗号（Searchable Encryption）」の研究が近年活発化している。検索可能暗号は、外部者にデータの内容を知られることなく、当該データの保管および検索を実施できる暗号であり、暗号化された大量のデータの保管やキーワード検索のアウトソース化を安全に実現するための技術として注目されている。最近では、検索可能暗号を活用したクラウド・サービスの提供も開始されている[4]（NTT ソフトウェア［2013］、日立ソリューションズ［2014, 2016b］）。

検索可能暗号は、そのアルゴリズムの種類によって、公開鍵暗号型と共通鍵暗号型の 2 つに分類される。公開鍵暗号型は、暗号化用の鍵と復号用の鍵が異なり、暗号化用の鍵を全てのエンティティに公開できるため、任意のエンティティがデータを預託できるという特徴を有している。共通鍵暗号型は、暗号化用の鍵と復号用の鍵が同一であり、各エンティティが鍵を秘密に保管する必要がある。そのため、この鍵を保有するエンティティのみがデータを預託できる一方、一般に、公開鍵暗号型と比較して、大量のデータをより速くキーワード検索できるという特徴を有している[5]。

今後、検索可能暗号を活用したクラウド・サービスが普及するようになれば、金融機関は、機密性の高いデータを自行内で暗号化したうえで外部事業者に預託しつつキーワード検索等を行うことが可能となり、外部事業者のセキュリティ管理の状況によらず、鍵管理を通じてデータの機密性にかかるリスクを制御することができるようになると期待される。本稿では、金融機関において、大量のデータを取り扱う業務をクラウド・サービスにアウトソース化する際に共通鍵暗号型の検索可能暗号（以下、特にことわらない限り、単に「検索可能暗号」という）を使用することを想定し、達成される安全性要件を揃えたうえで、最近の主要な実現方式について、処理性能の観点から評価・比較する[6]。

本稿の構成は以下のとおりである。2 節では、検索可能暗号のモデルを設定し、既存の検索可能暗号において提案されている検索機能の種類（キーワードが完全に合致しているデータのみを検索対象とするなど）と最近の主要な実現方式を説明する。3 節では、検索可能暗号の処理性能の評価尺度を示し、本稿で検討

[4] 100 万語の単語から構成される文書に対して、約 4 ミリ秒でキーワード検索を実行可能であり、平文状態での処理に近い検索スピードを達成するシステムが提案されている（日立ソリューションズ［2016a］）。また、1 万件規模のデータに対して、約 8 ミリ秒でキーワード検索が可能なシステムも提案されている（日立ソリューションズ［2012］）。

[5] 公開鍵暗号型の検索可能暗号の詳細については、清藤・四方［2014］、清藤・青野・四方［2017］等を参照されたい。

[6] 検索可能暗号の安全性については、太田［2017］を参照されたい。

対象とする評価尺度について説明する。4 節では、2 節で示した実現方式を対象に、クラウド・サービスに預託するデータの量（登録ファイル数）を変数として、各尺度の値の系列を算出・基準化し、それらを比較して相対的に望ましい方式について検討する。

２．検索可能暗号

（１）検索可能暗号のモデル

検索可能暗号の利用形態として、ここでは、「クラウド・サービスの利用者が、業務で作成したデータをクラウド・サービスの外部事業者に預託し、後日、キーワード検索を実行して、該当するデータをクラウド・サービスの外部事業者から抽出する」というものを想定する。本稿で検討対象とする検索可能暗号のモデルは、「クラウド・サービスの利用者（以下、「登録者」と呼ぶ）」と「クラウド・サービスを提供する外部事業者（以下、「サーバ」と呼ぶ）」の 2 つのエンティティから構成されるものとする[7]。

登録者は、サーバに預託するデータを生成するとともに、後日サーバに預託したデータを利用する。また、データの暗号化や復号等に用いる「秘密鍵」を生成する。検索可能暗号を用いたサービスを金融機関が利用する場合、登録者が金融機関に該当する。サーバは、ストレージ等の計算資源を有し、登録者から受信したデータを保管するとともに、登録者からの要求に応じて検索処理を実行し、その処理結果を登録者に送信する。

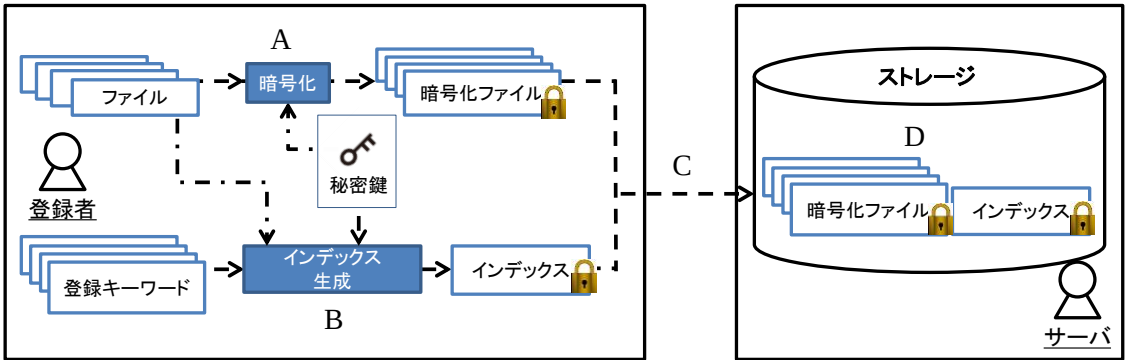
（２）3 つの処理フェーズ

上記のモデルにおいては、登録者が、(i) 秘密鍵を生成する処理（以下、「鍵生成フェーズ」と呼ぶ）、(ii) サーバに預託するデータ（以下、「ファイル」と呼ぶ）に対して暗号化等を実施し、そのうえでサーバに送信・預託する処理（以下、「登録フェーズ」と呼ぶ）、(iii) キーワード検索を行う処理（以下、「検索フェーズ」と呼ぶ）が想定される。

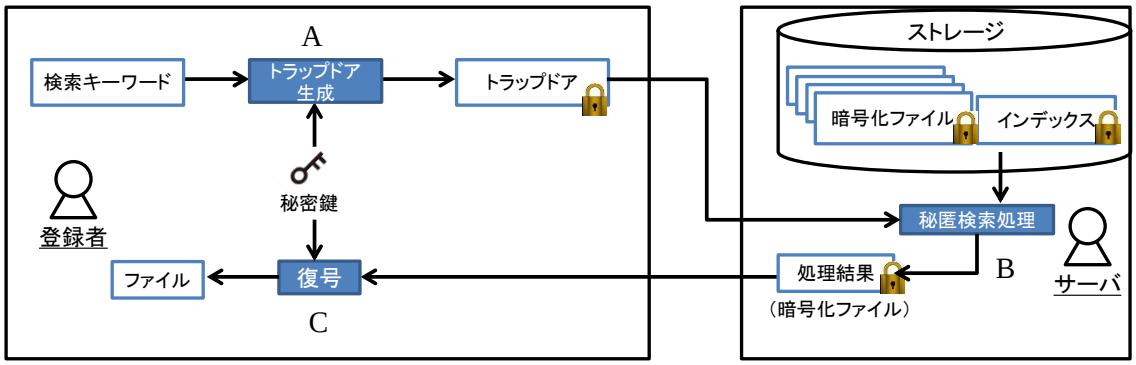
鍵生成フェーズでは、登録者は、秘密鍵を生成し、外部に漏えいしないように安全に保管する。登録フェーズでは、登録者は、まず、秘密鍵を用いてファ

[7] 現在提案されている検索可能暗号の中には、登録者とサーバに加えて、サーバに対して検索処理を要求して、キーワード検索を実行する「利用者」と呼ばれるエンティティを想定した、3 つのエンティティで構成されるモデルも研究されている（例えば、Curtmola *et al*. [2006]、Jerecki *et al*. [2013]、Ishai *et al.* [2016]）。ただし、そうしたモデルにおいては、登録者は利用者に対して暗号化と復号に使用する秘密鍵を共有することになる。したがって、秘密鍵の所持の観点からは、登録者と利用者は同一の情報を有する同一のエンティティであると捉えられるため、本稿のモデルと同一であると考えることができる。

図表 1　登録フェーズの処理フロー



図表 2　検索フェーズの処理フロー



イルを暗号化するとともに（図表 1－A）、ファイルやそれに含まれるキーワード（以下、「登録キーワード」と呼ぶ）を秘密鍵によって変換し、ファイルと登録キーワードとの対応関係を示すデータを暗号化した「インデックス」と呼ばれるデータを生成する（図表 1－B）。インデックスは、サーバに対してファイルや登録キーワードを秘匿したまま、サーバが効率的に検索処理を実行するために用いられる。次に、登録者は、暗号化ファイルとインデックスをサーバに送信し（図表 1－C）、サーバはこれらを保管する（図表 1－D）。検索フェーズにおいては、登録者は、入手したいファイルと対応付けされていると考えられるキーワード（以下、「検索キーワード」と呼ぶ）に対して秘密鍵を用いて一定の変換処理を施し（図表 2－A）、生成されたデータ（以下、「トラップドア」と呼ぶ）をサーバに送信する。サーバは、トラップドアとインデックスを用いて秘匿検索処理を実行し、トラップドアに対応する暗号化ファイルを抽出して登録者に返信する（図表 2－B）。登録者は、暗号化ファイルを自分の秘密鍵で復号する（図表 2－C）。

（３）安全性にかかる要件

イ．保護の対象とするデータ

本稿のモデルにおける保護の対象とするデータは、（暗号化されていない）ファイル、各ファイルに含まれる登録キーワード、検索キーワードである。一般に、想定される攻撃者（後述）に対して、①これらのデータが漏えいしない（機密性）、②データの改ざんを検知できる（完全性）ことが求められる。さらに、実際の利用を想定すると、上記①、②に加え、③必要なときにデータを利用できる（可用性）ことが求められる。

サーバに預託したデータの完全性を登録者が確認する方法としては、共通鍵暗号に基づくメッセージ認証コードを利用する方法等が知られている。また、可用性に関して、通常クラウド・サービスでは、サーバ等が冗長化されるほか、サービス・レベルの合意（Service Level Agreement: SLA）をアプリケーションに応じて適切に設定したり、障害発生時の対応を関係者間で事前に決定しておいたりすることなどを通じて、必要なレベルの可用性の確保に努めている（宇根・鈴木・吉濱［2011］）。本稿では、完全性と可用性に関しては、こうした対応が実施されていることを前提として検討の対象外とし、データの機密性の確保にかかる対応に焦点を当てて検討する。

ロ．攻撃者と安全性要件

本稿のモデルにおける安全性（データの機密性）の要件を整理する前に、その前提となる攻撃者を明確にしておく。ここでは、登録者とサーバとの間の通信路を観測する第三者のみならず、サーバの運用者が攻撃者となることを想定する[8]。攻撃者は、サーバの管理者権限等を奪取した後、登録者から預託されたデータの内容や対応関係[9]等の推測を試みる可能性がある。具体的には、サーバ内部で保管される暗号化ファイル、インデックスに加えて、トラップドア、暗号化ファイル等の保管や検索の処理の過程でサーバのメモリ等に現れるランタイム・データを攻撃に利用できるとする。

検索可能暗号では、その特性上、暗号化ファイルの数やサイズ、登録キーワードの数等の情報がサーバに知られることとなる[10]。本稿では、上記の検討を基に、以下の 3 点をデータの機密性にかかる安全性要件とする。想定する攻撃者に対して①ファイル（平文）が漏えいしない、②各ファイルに含まれる登録キーワー

---

[8] 登録者については、ファイルをサーバに預託する主体であり、不正行為を行わず、秘密鍵も安全に管理するとして検討を進める。

[9] 例えば、特定の登録キーワードが各ファイルに含まれる確率や、特定の登録キーワードが含まれるファイルに別の特定の登録キーワードが含まれる確率等が挙げられる。

[10] カートモラら（Curtmola *et al*. [2006]）は、これらよりも多くの情報が漏えいするか否かを基準に方式の安全性を定義しており、安全性評価の観点でも代表的な研究と目されている。

ド、および、検索キーワードが漏えいしない、③登録キーワード、あるいは検索キーワードとファイルとの間の対応関係が漏えいしない。

図表 3　検索可能暗号の主な種類

検索の種類			機能の概要	主な実現方式
単一キーワード方式	完全一致		検索キーワードと同一の登録キーワードに対応する暗号化ファイルを抽出する。例えば、検索キーワード「銀行」に対し、登録キーワード「銀行」に対応する暗号化ファイルのみを抽出する。	Curtmola *et al*. [2006] Yavuz and Guajardo [2015] Asharov *et al*. [2016] 黒澤ほか［2016］
	部分一致		検索キーワードの文字列を含む登録キーワードに対応する暗号化ファイルを抽出する。例えば、検索キーワード「銀行」に対し、「全国銀行協会」や「地方銀行」等の登録キーワードに対応する暗号化ファイルを抽出する。	平野ほか［2016］ 早坂ほか［2016］ Chase and Shen [2015] Faber *et al*. [2015]
	範囲		キーワード同士に順序が定義されている場合に、検索キーワードで指定する範囲に含まれる登録キーワードに対応する暗号化ファイルを抽出する。例えば、検索キーワード「2017 年 1 月 1 日～2017 年 3 月 31 日」に対し、その期間に含まれる日付を登録キーワードとする暗号化ファイルを全て抽出する。	Popa *et al*. [2011], Faber *et al*. [2015]
	類似（完全一致、部分一致、範囲以外）		検索キーワードと（類似性に関する）一定の関係（完全一致、部分一致、範囲検索以外）を有する登録キーワードに対応する暗号化ファイルを抽出する。例えば、検索キーワード「取引」に対し、「取引き」や「取り引き」等の登録キーワードに対応する暗号化ファイルを抽出する。	海上ほか［2016］ Li *et al.* [2010] Boldyreva and Chenette [2014]
複数キーワード方式	完全一致	論理積	複数の検索キーワードを指定し、それらがすべて登録キーワードとして対応する暗号化ファイルを抽出する。例えば、2 つの検索キーワード「銀行」と「取引」に対し、登録キーワードとして「銀行」と「取引」の両方を含む暗号化ファイルを抽出する。	小嶋ほか［2016］
		論理和	複数の検索キーワードを指定し、それらのうち少なくとも 1 つが検索キーワードとして対応する暗号化ファイルを抽出する。例えば、2 つの検索キーワード「銀行」と「取引」に対し、登録キーワードとして「銀行」または「取引」のいずれかを含む暗号化ファイルを抽出する。	Cash *et al*. [2013] Kurosawa [2014] Gajek [2016]
		論理積と論理和の組合せ	論理積検索と論理和検索を組み合わせて検索する。例えば、「地方銀行」と「都市銀行」の両方を含む、または「取引き」を含むなどの条件でキーワード検索を実行する。	Cash *et al*. [2013] Kurosawa [2014] Gajek [2016]

（４）検索機能の整理

既存の提案方式において実現される主な検索機能（図表 3 参照）は、1 つのキーワードを用いるもの（以下、「単一キーワード方式」という）と複数のキーワードを用いるもの（以下、「複数キーワード方式」という）に大別される。さらに、単一キーワード方式は、「完全一致検索」と「部分一致検索」、「類似検索[11]」、「範囲検索」に分けられるほか、複数キーワード方式は、「論理積（AND）検索」、「論理和（OR）検索」、「論理積検索と論理和検索の組合せ」に分けられる。

図表 3 の「主な実現方式」で示したものはいずれも「少なくとも上記の安全性要件①、②、③を満たす」ことから、一定レベル以上の安全性を満足する方式として、4 節において横並びで性能評価を行うこととする[12]。

３．処理性能の評価方法

本節では、まず、検索可能暗号の処理性能の評価尺度について説明する。次に、評価・比較の際に着目するパラメータについて説明し、評価する際のパラメータの値を設定する。そのうえで、2 節（4）で述べた検索機能ごとに、具体的な実現方式の処理性能を評価・比較する。

（１）評価尺度

処理性能を評価する尺度は、一般に、データの「処理量」に関する尺度と、データの「大きさ（サイズ）」に関する尺度に分けられる（図表 4）。

処理量に関する尺度としては、①インデックス生成の処理（図表 1－B）にかかる計算量（以下、「インデックス生成処理量」という）、②トラップドア生成の処理（図表 2－A）にかかる計算量（以下、「トラップドア生成処理量」という）、③検索処理（図表 2－B）にかかる計算量（以下、「検索処理量」という）がある。検索処理量に関しては、より具体的な処理にかかる評価尺度として、④「局所性（locality）」と⑤「読取効率（read efficiency）」が存在する。局所性は、メモリ領域上で物理的に離れて保管されているデータをそれぞれ抽出する際に、当該データの（先頭）アドレスへのアクセス回数として定義される。読

[11] 検索キーワードの類似文字列を予め生成し、それらすべてに対してトラップドアを生成し、完全一致検索を繰り返して、対応する暗号化ファイルを抽出する方式も含む。なお、このような実現方式の処理性能は、ベースとなる完全一致検索の実現方式の処理性能に依存する（例えば、尾形ほか［2015］を参照）。

[12] 例えば、本節（3）ロ．で想定した攻撃に加えて、特定のデータを用いずに何らかの情報を積極的に得ようとする攻撃（秘密鍵を使わずに生成したトラップドアに対する検索結果から推測を試みる攻撃等）を想定したうえで、安全性要件①、②、③を満たすなど、より強い安全性を実現する方式も存在する（例えば、黒澤ほか［2016］）。

図表 4　処理性能に関する主な評価尺度

評価尺度			概要
処理量	インデックス生成処理量		登録者が、登録フェーズにおいて、インデックスの生成処理に要する計算量
	トラップドア生成処理量		登録者が、検索フェーズにおいて、トラップドアの生成処理に要する計算量
	検索処理量		サーバが、検索フェーズにおいて、受信したトラップドアに応じて実施する検索処理に要する計算量
		局所性	サーバが、検索処理をする際に、メモリ上で物理的に離れた先頭アドレスへの（必要な）アクセス回数
		読取効率	サーバが、正しい検索結果として出力するデータのサイズに対する、メモリ上から読み出す必要があるデータのサイズの比率
サイズ	インデックスのサイズ		登録者が生成したインデックスのサイズ
	トラップドアのサイズ		登録者が生成したトラップドアのサイズ

備考：これらの尺度の評価・比較については補論の図表 A-2 を参照されたい。

取効率は、正しい検索結果として得られるデータのサイズに対する、サーバがトラップドアに応じてメモリ領域から抽出するデータのサイズの比率である（Cash and Tessaro [2014]）。なお、ファイルの暗号化（図表 1－A）と復号（図表 2－C）には、AES 等の共通鍵暗号が用いられ、これらの処理にかかる計算量は、共通鍵暗号の種類やファイルサイズに依存し、検索可能暗号の実現方式に依存しないため、ここでは検討対象外とする[13]。

サイズに関する尺度としては、インデックスのサイズとトラップドアのサイズが挙げられる。一般に、インデックスのサイズは、登録者とサーバ間の通信量とサーバのストレージに影響し、トラップドアのサイズは、通信量のみに影響する。検索可能暗号を利用する際には、サーバが暗号化ファイルとインデックスを保管するためのストレージが必要となる。暗号化ファイルのサイズは、ファイルサイズに依存し、実現方式に依存しないため、ここでは検討対象外とする。

（２）本稿における評価の観点

本稿では、「登録フェーズや検索フェーズにおいて登録するファイルの数（以下、「登録ファイル数」という）が増加した場合に、計算量やメモリ領域がどの

[13] 処理量にかかる尺度のうち、暗号化／復号にかかる処理量（以下、「暗号化処理量」という）が、他の処理量に比べて著しく大きい場合、暗号化処理量以外の尺度に着目する意味はない。もっとも、暗号化処理量が他の処理量と比べてどの程度大きいかは方式に依存し自明とはいえないことから、暗号化処理量以外の尺度を検討対象として議論を進める。

程度の割合で増加するか」に焦点を当てて評価・比較することとする。1 節で述べたように、検索可能暗号では、大量のデータを効率よく検索・抽出できるという利点がある。こうした利点を活かして検索可能暗号を採用することを検討する際には、将来のビジネス上のニーズや業務範囲の拡大等に伴って、より多くのデータの預託や検索が必要になった場合に、計算量やメモリ領域等のリソースをなるべく節約することができる方式が相対的に望ましいといえる。こうした考え方に基づき、以下では、登録ファイル数をある値に固定して考えるのではなく、登録ファイル数が増加した際に、計算量やメモリ領域の評価尺度がどのように増加するかを明確にしたうえで、各評価尺度の値を各方式間で比較することとする。

上記の観点での評価においてどの尺度を利用するかに関しては、すべての尺度について評価するのではなく、尺度間の関係や傾向に着目し、複数の尺度のなかから代表的なものを選択して用いることが効果的である。その代表的な評価尺度として、インデックスのサイズ、トラップドアのサイズ、検索処理量の 3 つを選択することで、図表 4 で示した尺度を網羅することができる。

まず、登録フェーズにおいて、登録者がファイルを暗号化してインデックスとともにサーバに送信する際、インデックスのサイズが増加すると、それに伴いインデックス生成処理量（および、データ登録にかかる時間）も増加することから、インデックスのサイズを代表的な尺度とする。

次に、検索フェーズにおいて、登録者による検索の頻度が高まると、一定時間当りのトラップドアのサイズが増加するほか、それに伴いトラップドア生成処理量（および、その結果として、検索結果を得るまでにかかる時間）も増加する。このため、トラップドアのサイズを代表的な尺度とする。一方、検索処理量については、各実現方式における検索アルゴリズムに依存し、インデックスやトラップドアのサイズによって代表させることができない。

（3）評価する際の具体的なパラメータ値

各評価尺度の値は、登録ファイル数等のさまざまなパラメータに依存して定まる。各パラメータの値が同じであるという条件のもとで、評価尺度の値が小さい方が処理性能の観点から望ましいと考えられる。各実現方式の処理性能の評価・比較を行うために、（登録ファイル数以外の）いくつかのパラメータを固定して考える。

まず、各ファイルに含まれる登録キーワードの数の平均値（以下、「登録キーワード数」という）を設定する。キャッシュら（Cash *et al.* [2013]）は、検索可能暗号の適用に関する実装評価の結果、および、実装評価用のデータセットの値を発表しており、約 150 万件のファイルに対して、当該ファイルと登録キー

ワードのペアを約 1 億 5 千万件準備している[14]。したがって、登録ファイル数は 100 となるため、本稿においても登録キーワード数を 100 とする。

秘密鍵のサイズは、電子政府推奨暗号リストにおいて当面は安全であるとされる共通鍵暗号（AES 等）の鍵長を基に、128 ビットとする（CRYPTREC［2013］）。さらに、個々のファイルの暗号化にかかる時間は、暗号化するファイルのサイズに依存して変化するため、その中で最大のものを「1 つのファイルの暗号化にかかる時間」とする。その他、預託する暗号化ファイルの識別子のビット長を 32 ビット、ハッシュ関数（SHA-256 等）の出力値のビット長を 256 ビットとする[15]。

４．検索可能暗号の処理性能の評価・比較

ここでは、各方式の処理性能について、以下の手順により評価・比較を行う。各尺度の評価式（補論の図表 A-2）に具体的なパラメータを代入し、各尺度の値が登録ファイル数の増加に対してどのように変化するかを方式ごとに見積り比較する。

（１）完全一致検索の実現方式の性能評価

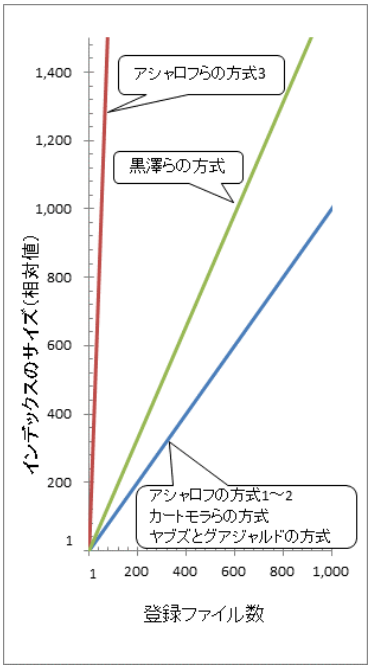
各検索機能を実現する方式のうち、比較的最近（2010 年以降）提案された代表的なものを性能評価の対象とする。すなわち、ヤブズとグアジャルドの方式（Yavuz and Guajardo [2015]）、アシャロフらの提案した 3 つの方式（以下、「アシャロフらの方式 1～3」という、Asharov *et al.* [2016]）、黒澤らの方式（黒澤ほか［2016］）を対象とする。さらに、初期の代表的な研究と目されるカートモラらの方式（Curtmola *et al*. [2006]）を加えて、合計で 6 つの方式を評価対象とする。

---

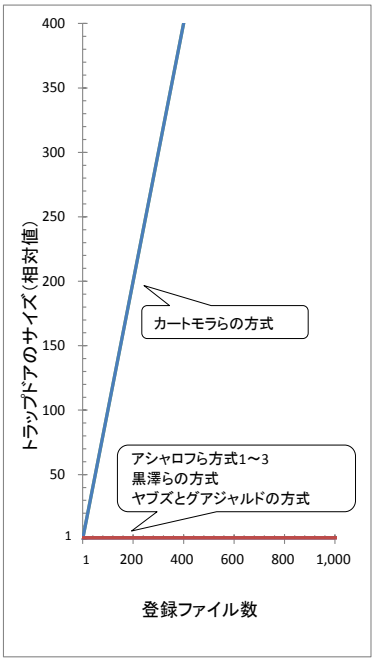
[14] キャッシュら（Cash *et al.* [2013]）による実装評価における検索処理にかかる時間の計測値から、登録ファイル数を 100 万としたときの完全一致検索のキーワード検索処理にかかる時間は、高々1 秒程度であると考えられる。なお、キャッシュらは、実装評価に際して、電子メールのデータセットを利用している。

[15] これらの比較のベースとなる評価尺度とパラメータの関係（メモリ・サイズや計算量のオーダー）、その他の評価尺度に影響を与えるパラメータの詳細については、補論を参照されたい。また、黒澤ら（黒澤ほか［2016］）の方式では、「サーバが悪意を持って不正な検索結果を出力したときに、登録者が当該不正行為を検出する」機能を有しており、登録者が暗号化ファイルとインデックスに加えて、検索結果の不正な操作を検知するためのデータを別途準備してサーバに預託するという処理を組み込んでいる。こうした処理にかかるパラメータとして、サーバによる不正な操作を検知するためのデータのサイズ（整数サイズ）を 2,048 ビットとしている。

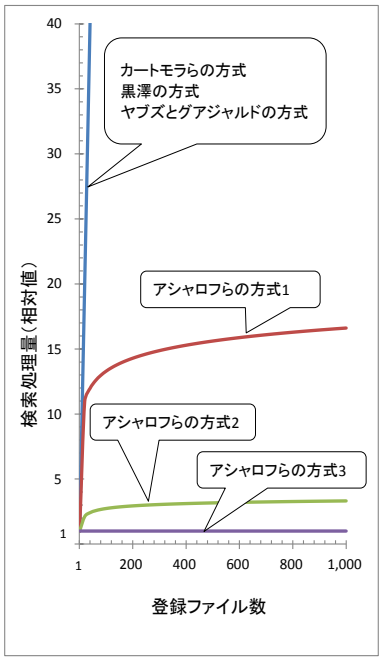
図表 5 完全一致検索の実現方式の性能評価



①インデックスのサイズによる比較



②トラップドアのサイズによる比較



③検索処理量による比較

3 節（2）で示したインデックスのサイズ、トラップドアのサイズ、検索処理量が、各実現方式において、登録ファイル数の増加に対してどのように増加するかを示したのが図表 5 である。図表 5 の横軸は登録ファイル数を示しており、登録ファイル数を「1」としたときの各尺度の値を方式ごとに計算し、そのなかでの最小値を「1」として各尺度の値を基準化する。そして、横軸の値の変化に伴い、各尺度の系列（基準化されたもの）がどう変化するかを示す。登録ファイル数を同じ値としたとき、評価尺度の値が小さい方が処理性能の観点から相対的に望ましい方式であると考えられる。例えば、インデックスのサイズの値が相対的に小さい方が、登録にかかる時間が短くなると見積ることができる。

インデックスのサイズに関しては（図表 5 の①）、いずれの方式も、登録ファイル数が 1 であるときの値はほぼ同じであるが、アシャロフらの方式 3 では、インデックスのサイズが、（単なる登録ファイル数でなく）「“登録ファイル数（N）”と、“登録ファイル数の自然対数（$\log N$）”の積」（すなわち、$N \log N$）に比例して増加することから、インデックスのサイズの増加の割合が相対的に大きい。その他の方式では、インデックスのサイズが登録ファイル数に対して 1 次関数的に増加するが、それらの中でも、黒澤らの方式において、登録ファイル数に対するインデックスのサイズの増分が相対的に大きい。トラップドアのサイズに関しては（図表 5 の②）、カートモラらの方式では、登録ファイル数の増加に対して 1 次関数的に増加する。その他の方式では、登録ファイル数に依存せず

一定となり、トラップドアの生成処理にかかる時間等を抑えられる方式であると考えられる。検索処理量に関しては（図表 5 の③）、アシャロフらの方式 3 では、登録ファイル数に依存せず一定となる。また、アシャロフらの方式 1 では、登録ファイル数（N）の増加に対して対数関数的（$\log N$）に増加するほか、アシャロフらの方式 2 では、さらにその対数関数的（$\log\log N$）に増加する。その他の方式では、登録ファイル数の増加に対して 1 次関数的に増加する。

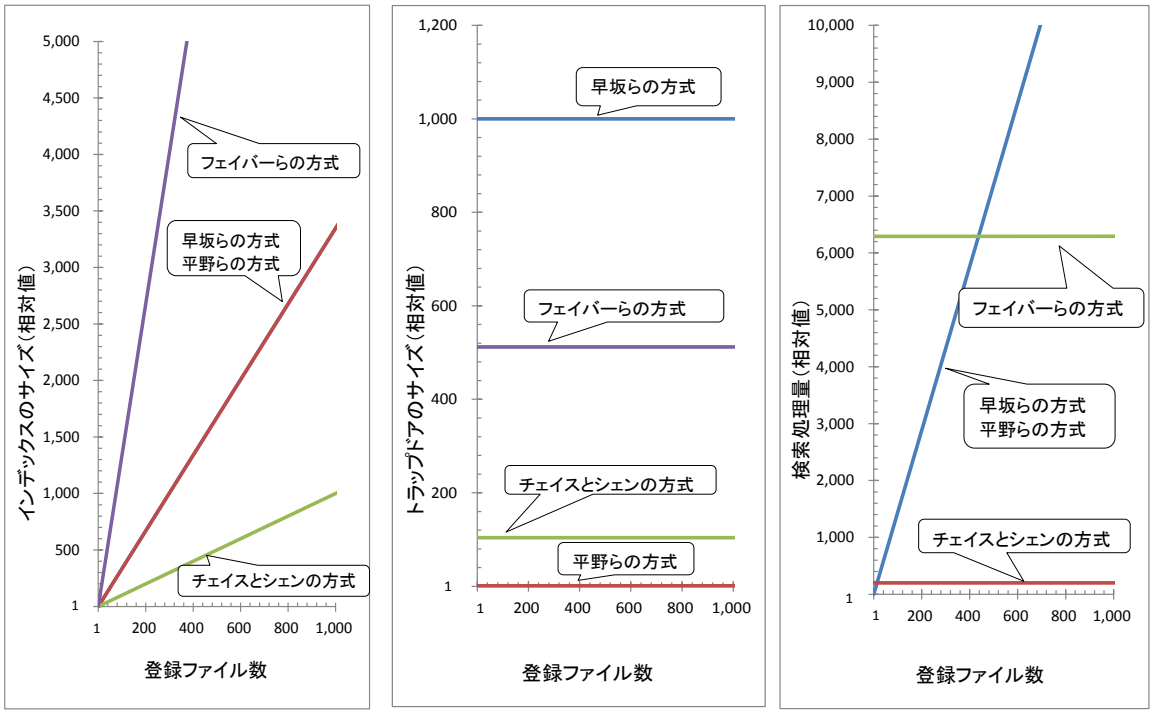
このようにみていくと、全ての尺度において相対的に望ましい方式は、評価対象の中には存在しない。インデックスのサイズに焦点を当てると、アシャロフらの方式 3 以外の方式が相対的に望ましく、トラップドアのサイズに焦点を当てると、カートモラらの方式以外の方式が相対的に望ましいといえる。検索処理量に焦点を当てると、アシャロフらの方式 3 が相対的に望ましいといえる。インデックスのサイズよりも、トラップドアのサイズや検索処理量を優先的に抑制することが求められる用途（例えば、サーバにおける検索処理の高速化を優先したいというケース）においては、アシャロフらの方式 3 を選択することが考えられる[16]。また、検索処理量よりも、インデックスのサイズとトラップドアのサイズを優先的に抑制することが求められる用途（例えば、登録者の端末の性能に制限があり、インデックスのサイズやトラップドアのサイズを軽減したいというケース）においては、アシャロフらの方式 1～2 を選択することが考えられる。

（２）部分一致検索の実現方式の性能評価

完全一致検索の実現方式は、登録キーワードと検索キーワードが完全に一致する場合のみを対象とする方式であり、登録キーワードの一部が不明な状況を想定すると、こうした方式の利用が有効であると考えられる。ここでは、最近提案された主要な方式として、早坂らの方式（早坂ほか［2016］）、平野らの方式（平野ほか［2016］）、チェイスとシェンの方式（Chase and Shen [2015]）、フェイバーらの方式（Faber *et al*. [2015]）を対象とする。

[16] 例えば、ファイルの識別子のサイズを 32 ビット、登録キーワード数を 100、登録ファイル数を 10 万とした場合、アシャロフらの方式 1～2 でのインデックスのサイズは 40 メガバイト（の定数倍）程度、アシャロフらの方式 3 でのインデックスのサイズは 930 メガバイト（の定数倍）程度となる見積りである。

図表 6　部分一致検索の実現方式の性能評価



①インデックスのサイズによる比較　②トラップドアのサイズによる比較　③検索処理量による比較

インデックスのサイズに関しては、どの方式も登録ファイル数に対して 1 次関数的に増加するが、チェイスとシェンの方式はその増加の度合いが比較的小さい（図表 6）。トラップドアのサイズに関しては、登録ファイル数に依存する方式は性能評価の対象の中には存在しない[17]。検索処理量に関しては、チェイスとシェンの方式は登録ファイル数に依存しない[18]。その他の方式は登録ファイル数に対して 1 次関数的に増加するが、それらのうち、フェイバーらの方式は、増加の度合いが比較的緩やかである。

このように、3 つの尺度全てにおいて相対的に望ましい方式は、評価・分析の対象の中には存在しない。トラップドアのサイズに焦点を当てると、平野らの方式が相対的に望ましい。インデックスのサイズ、あるいは、検索処理量に焦点を当てると、チェイスとシェンの方式が望ましい。

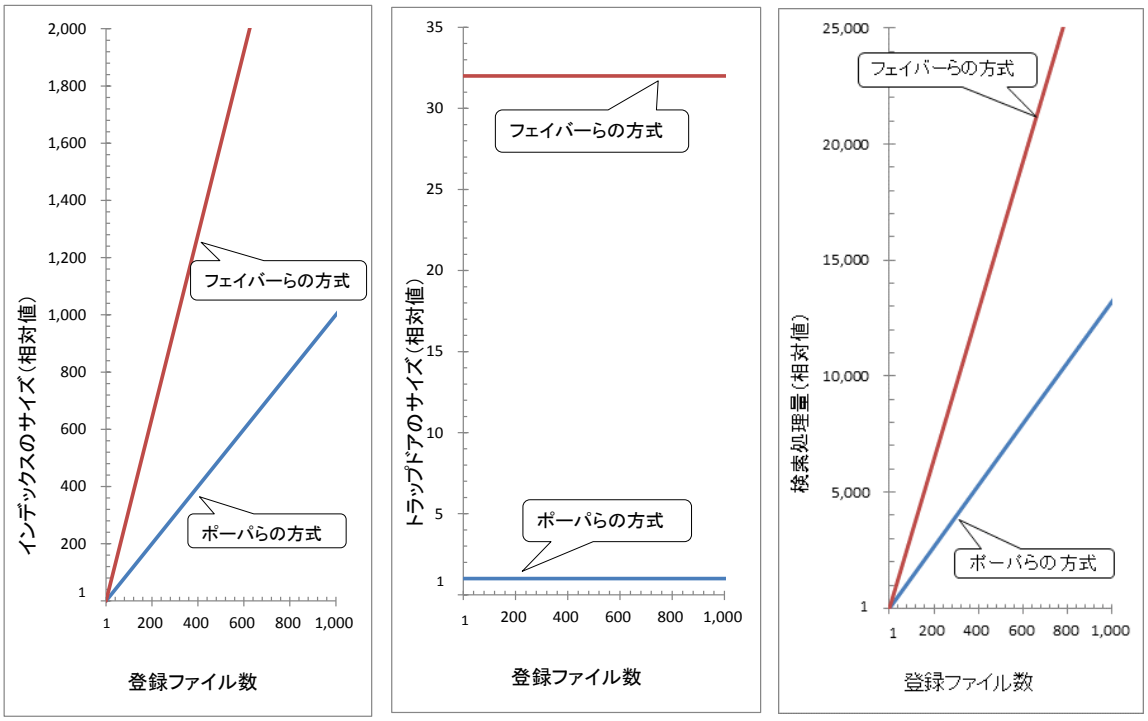
（３）範囲検索の実現方式の性能評価

キーワードが日付のように順序付けられている場合に、ある特定の期間を検索対象とすることを想定する。単純に対象の日付全てを検索キーワードとして

[17] 早坂らの方式におけるトラップドアのサイズは、全キーワード数に依存する。チェイスとシェンの方式におけるトラップドアのサイズは、検索キーワード長と検索ヒット数に依存する。フェイバーらの方式におけるトラップドアのサイズは、検索キーワード長と整数サイズに依存する。

[18] チェイスとシェンの方式では、検索処理量は検索ヒット数と検索キーワード長に依存する。

図表 7　範囲検索の実現方式の性能評価



①インデックスのサイズによる比較　②トラップドアのサイズによる比較　③検索処理量による比較

完全一致検索を実行するよりも、範囲検索の実現方式では、効率よく検索処理を実行可能である。これを実現する方式のうち最近提案されたポーパらの方式（Popa *et al.* [2011]）とフェイバーらの方式（Faber *et al.* [2015]）を性能評価の対象とする（図表 7）。

インデックスのサイズに関しては、両方式とも登録ファイル数に対して 1 次関数的に増加するが、ポーパらの方式の増加の度合いが比較的小さい。トラップドアのサイズに関しては、両方式とも登録ファイル数に依存せず一定の値となるが、ポーパらの方式において相対的に小さくなる[19]。検索処理量に関しては、フェイバーらの方式では、登録ファイル数に対して 1 次関数的に増加する一方で、ポーパらの方式は登録ファイル数（N）とその対数（$\log N$）の積（$N \log N$）に比例して増加する。このため、登録ファイル数が比較的小さい場合には、ポーパらの方式における検索処理量の増加の度合いが比較的小さいものの、登録ファイル数が大きくなると、フェイバーらの方式の方が増加の度合いが小さくなる[20]。

[19] フェイバーらの方式におけるトラップドアのサイズの最大値は、登録キーワード長の定数倍で抑えられる。

[20] 例えば、登録キーワード長を 32 ビットとすると、これらの方式における検索処理量が逆転するのは登録ファイル数が数十億以上（登録キーワード長が 64 ビットの場合は、その数十億倍）となる場合であり、こうした超大量のファイルを取扱う場合を除いて、実運用上はポーパらの方

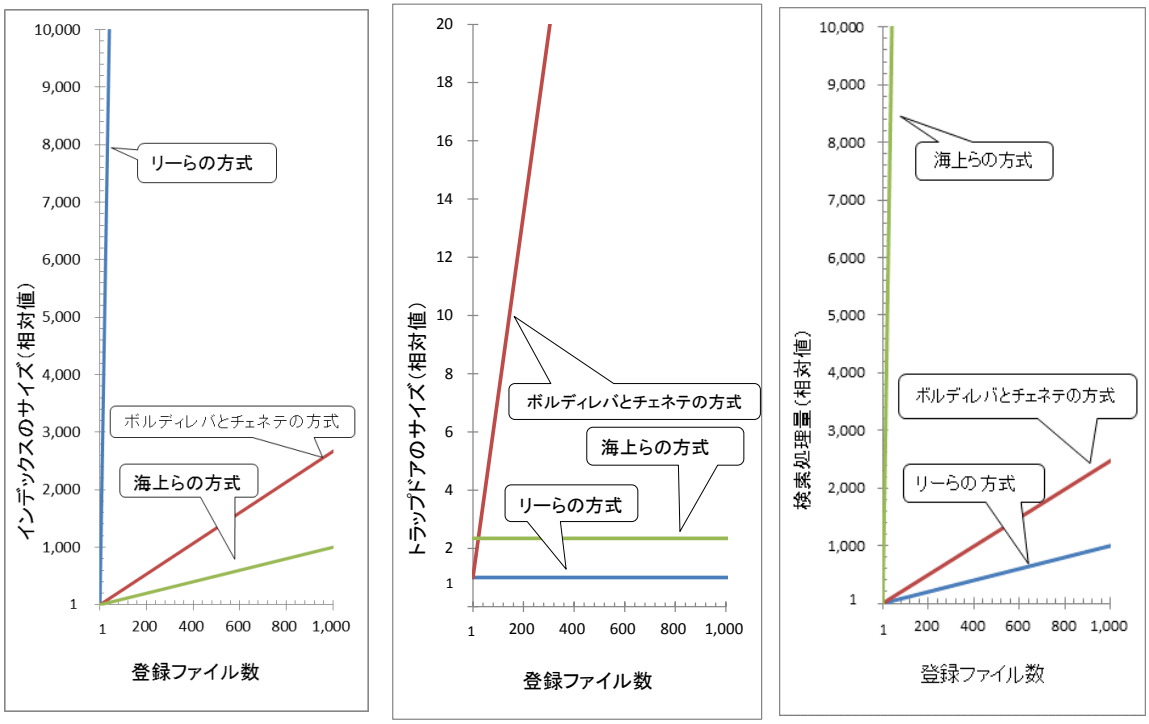
3 つの尺度全てにおいて相対的に望ましい方式は、評価・分析の対象の中には存在しない。インデックスのサイズとトラップドアのサイズの両方に焦点を当てると、ポーパらの方式が相対的に望ましい。一方、検索処理量に焦点を当てると、登録ファイル数の多寡に応じて相対的に望ましい方式が変わってくる。

（4）類似検索の実現方式の性能評価

検索キーワードの設定ミスや生体認証等、検索キーワードに相当する値に誤差が生じる状況を想定すると、類似検索の実現方式が有効である。これを実現する主要な方式のうち最近提案されたリーらの方式[21]（Li *et al.* [2010]）、ボルディレバとチェネテの方式（Boldyreva and Chenette [2014]）、海上らの方式（海上ほか［2016］）を対象とする[22]（図表 8）。

インデックスのサイズに関しては、どの方式も登録ファイル数に対して 1 次関数的に増加するが、海上らの方式における増加の度合いが比較的緩やかであ

図表 8　類似検索の実現方式の性能評価



①インデックスのサイズによる比較　②トラップドアのサイズによる比較　③検索処理量による比較

式の方が検索処理量は少ないと考えられる（補論の図表 A-3）。

[21] リーらの方式は、予め検索キーワードの入力ミス等による誤差のレベルを指定したうえで、その誤差を表現するための登録キーワードを大量に用意して完全一致検索を行うというアイデアに基づく方式である。

[22] リーらの方式は検索キーワードの入力ミス等を想定しているのに対し、海上らの方式は、キーワードとなる情報に微小な揺らぎが存在（例えば、生体情報を取扱う場合）する状況を想定しており、想定するアプリケーションが異なっている。

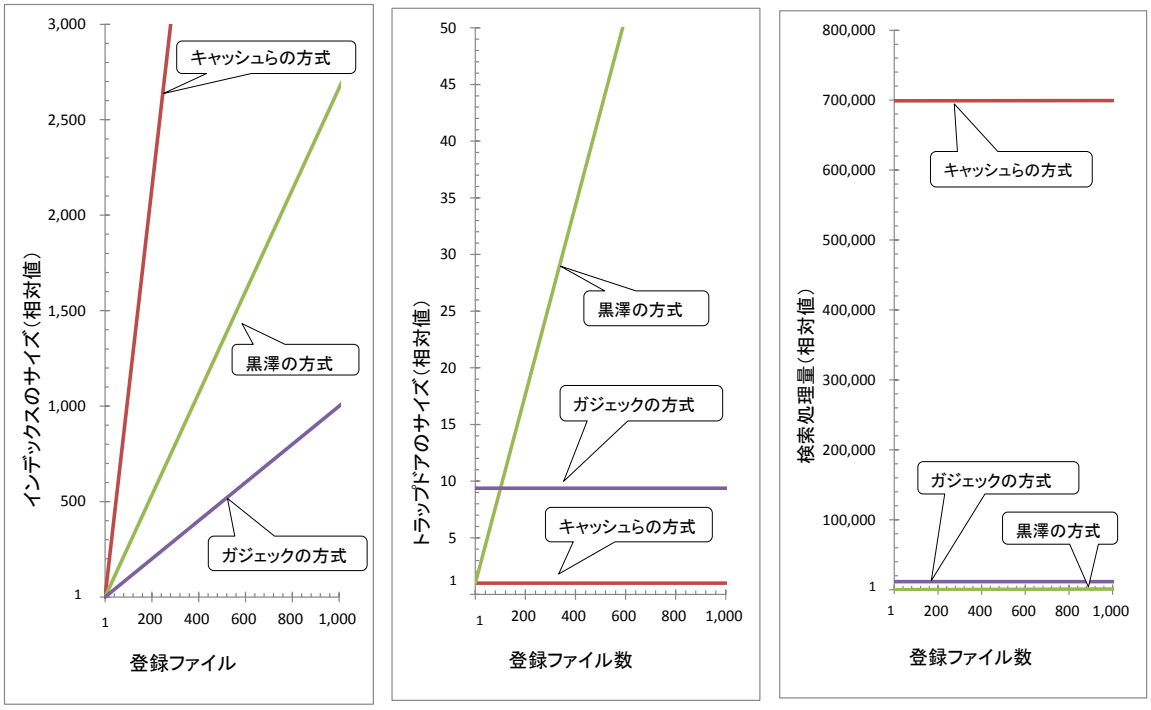
る。トラップドアのサイズに関しては、リーらの方式と海上らの方式は、登録ファイル数に依存しない[23]。一方、ボルディレバとチェネテの方式は、登録ファイル数に対して 1 次関数的に増加する。検索処理量に関しては、どの方式も登録ファイル数に対して 1 次関数的に増加するが、それらのうち、リーらの方式における増加の度合いが比較的緩やかである。

このように、インデックスのサイズに焦点を当てると、海上らの方式が相対的に望ましい。トラップドアのサイズや検索処理量に焦点を当てると、リーらの方式が望ましい。

（５）複数キーワード方式の実現方式の性能評価

大量のファイルの中から、複数のキーワードを指定し、精密な検索結果を効率的に取得したい場合、複数のキーワードを組み合わせて一度で検索する機能が有効である。ここでは、論理積と論理和の組合せ検索に焦点を当てて、最近提案された主要な方式である、キャッシュらの方式（Cash *et al.* [2013]）、黒澤の方式（Kurosawa [2014]）、ガジェックの方式（Gajek [2016]）を対象とする（図表 9）。

図表 9　複数キーワード方式の性能評価



①インデックスのサイズによる比較　②トラップドアのサイズによる比較　③検索処理量による比較

[23] トラップドアのサイズは、リーらの方式では検索キーワード長と置換回数に依存するほか、海上らの方式では登録キーワード数と登録キーワード長に依存する。

インデックスのサイズに焦点を当てると、いずれの方式も 1 次関数的に増加し、ガジェックの方式が相対的に望ましい。トラップドアのサイズに焦点を当てると、ガジェックの方式とキャッシュらの方式において登録ファイル数が増加しても一定となり、相対的にトラップドアのサイズが小さくなるキャッシュらの方式が望ましい[24]。検索処理量に焦点を当てると、キャッシュらの方式と黒澤の方式では、登録ファイル数に対して 1 次関数的に増加し、ガジェックの方式では一定となる。登録ファイル数が比較的小さい場合には、黒澤の方式における検索処理量が相対的に小さくなるが、登録ファイル数が相応に大きくなるとガジェックの方式の方が検索処理量が小さくなる[25]。

５．おわりに

本稿では、今後、クラウド・サービスにおいて実装される可能性がある共通鍵暗号型の検索可能暗号に焦点を当てて、検索可能暗号のモデルを設定し、実現可能な検索機能を整理した。そのうえで、処理性能の観点から評価する際に用いられる尺度を示すとともに、求められる安全性を揃えたうえで、検索機能ごとの各実現方式の評価・比較を行った。特に、同一の検索機能を有する最近の実現方式について、登録ファイル数を変数としたときの各尺度の値の系列に着目し、どの方式が相対的に望ましいかを示した。

最も基本的な検索機能である完全一致検索を実現する方式に関しては、インデックスのサイズ、トラップドアのサイズ、検索処理量の観点で、比較的高い処理性能を達成する方式が提案されている。こうした方式については、今後、実サービスへの適用を展望した実装にかかる研究開発の進展が期待される。その他の検索機能の実現方式に関しては、処理性能においてさらなる改良の余地が残されているといえる。例えば、部分一致検索を実現する方式の選択肢の 1 つとして考えられるチェイスとシェンの方式には、検索フェーズにおいて登録者とサーバ間で複数回のデータのやり取りが必要となっており、トラップドアのサイズ、トラップドアのサイズ等の観点でさらなる処理性能の向上が期待される。

現在、検索可能暗号の研究・開発が活発に行われており、より高い処理性能を有する検索可能暗号の実現方式が今後も提案されていくと考えられる。同時

[24] ガジェックの方式ではトラップドアのサイズと検索処理量は、登録キーワード数、登録キーワード長、検索キーワード数に依存する。キャッシュらの方式ではトラップドアのサイズは検索キーワード数と整数サイズに依存する。

[25] 登録ファイル数が 1 万以上の場合、ガジェックの方式における検索処理量は、黒澤の方式よりも相対的に小さくなると見積られる（補論の図表 A-4）。

に、こうした優れた実現方式を実装するクラウド・サービスも提供されるようになる可能性もある。金融機関においては、そうした新しいサービスの活用を展望しつつ、データの機密性を確保しながら業務の一層の効率化に資する技術として、検索可能暗号の動向を今後フォローしていくことが有用であろう。

以　上

参考文献

アマゾン・ウェブサービス（AWS）、「AWS 導入事例：株式会社　三菱 UFJ トラスト投資工学研究所」、2016 年（https://aws.amazon.com/jp/solutions/case-studies/mtec/）

宇根正志・鈴木雅貴・吉濱佐知子、「クラウド・コンピューティングにおける情報セキュリティ管理の課題と対応」、『金融研究』第 30 巻第 1 号、日本銀行金融研究所、2011 年、227～252 頁

海上勇二・松崎なつめ・山田翔太・アッタラパドゥンナッタポン・松田隆宏・花岡悟一郎、「ユークリッド距離に基づく類似検索可能暗号」、2016 年暗号と情報セキュリティシンポジウム発表資料、電気情報通信学会、2016 年

NTT ソフトウェア、「クラウド向けデータ暗号化・ログ監査ソリューション TrustBind/Secure Gateway」、2013 年（https://www.ntts.co.jp/products/ trustbind/sgw/）

太田和夫、「共通鍵暗号による秘匿検索暗号のセキュリティ」、『金融研究所ディスカッション・ペーパー・シリーズ』2017-J-5、日本銀行金融研究所、2017 年

尾形わかは・金岡晃・松尾真一郎、「実用的な多機能検索可能暗号方式～身も蓋もない方式を考えてみた～」、2015 年暗号と情報セキュリティシンポジウム発表資料、電気情報通信学会、2015 年

金融情報システムセンター、『金融機関等コンピュータシステムの安全対策基準・解説書（第 8 版、追補改訂）』、金融情報システムセンター、2015 年

――――、『金融情報システム増刊 80 号：平成 28 年度金融機関アンケート調査結果』No.341、金融情報システムセンター、2016 年

CRYPTREC、「CRYPTREC 暗号リスト」、CRYPTREC、2013 年

黒澤馨・佐々木圭祐・太田清比古・米山一樹、「UC 安全性を満たす効率的で動的な検索可能暗号」、2016 年暗号と情報セキュリティシンポジウム発表資料、電気情報通信学会、2016 年

小嶋陸広・品川和雅・金山直樹・西出隆志・岡本栄司、「共通鍵完全準同型暗号を用いた安全なブルームフィルタ」、2016 年暗号と情報セキュリティシンポジウム発表資料、電気情報通信学会、2016 年

清藤武暢・青野良範・四方順司、「公開鍵暗号型の高機能暗号を巡る研究動向」、『日本銀行金融研究所ディスカッション・ペーパー・シリーズ』2017-J-8、日本銀行金融研究所、2017 年

――――・四方順司、「高機能暗号を活用した情報漏えい対策「暗号化状態処理技術」の最新動向」、『金融研究』第 33 巻第 4 号、日本銀行金融研究所、2014

年、97～132 頁
早坂健一郎・川合豊・平野貴人・太田和夫・岩本貢、「共通鍵暗号型の秘匿部分一致検索（その２）」、2016 年暗号と情報セキュリティシンポジウム発表資料、電気情報通信学会、2016 年
日立製作所、「千葉銀行、日立がクラウドで提供する ATM ジャーナル集中管理サービスを導入し稼働を開始」、2013 年（http://www.hitachi.co.jp/New/ cnews/month/2013/03/0328.html）
――――、「山陰合同銀行、日立がクラウドで提供する ATM ジャーナル集中管理サービスを導入し稼働を開始」、2014 年（http://www.hitachi.co.jp/New/cnews/month/2014/04/0416.html）
日立ソリューションズ、「クラウド上での情報漏えい防止に貢献する検索可能暗号技術を開発　暗号化したゲノムデータベースの検索に応用」、2012 年（http://www.hitachi-solutions.co.jp/company/press/news/2012/0312.html）
――――、「NCNP と日立ソリューションズが「Remudy WEB 患者情報登録システム」の運用を開始」、日立ソリューションズ、2014 年（http://www.hitachi-solutions.co.jp/company/press/news/2014/1125.html）
――――、「Credeon Secure Full-text Search」、2016 年 a（http://www.hitachi-solutions.com/securesearch/、2016 年 12 月 16 日）
――――、「SharePoint Online 上の重要情報を守るセキュリティ強化ソリューションを提供開始クラウド上にアップロードされるすべての情報を暗号化し、第三者による不正参照を防止」、2016 年 b（http://www.hitachi-solutions.co.jp/company/press/news/2016/0829.html）
平野貴人・川合豊・太田和夫・岩本貢、「共通鍵暗号型の秘匿部分一致検索（その１）」、2016 年暗号と情報セキュリティシンポジウム発表資料、電気情報通信学会、2016 年
Asharov, Gilad, Moni Naor, Gil Segev, and Ido Shahaf, “Searchable Symmetric Encryption: Optimal Locality in Linear Space via Two-Dimensional Balanced Allocations,” *Proceedings of the 48th Annual ACM Special Interest Group on Algorithms and Computation Theory Symposium on Theory of Computing*, 2016, pp. 1101-1114.
Boldyreva, Alexandra, and Nathan Chenette, “Efficient Fuzzy Search on Encrypted Data,” *Fast Software Encryption*, Lecture Notes in Computer Science 8540, Springer-Verlag, 2014, pp. 613-633.
Box, “Nonprofit Partnerships and Success Stories,” Box, 2016 (https://www.box.org/success-stories#node-101).
Cash, David, Stanislaw Jarecki, Charanjit Jutla, Hugo Krawczyk, Marcel Rosu, and

Michael Steiner, “Highly-Scalabe Searchable Symmetric Encryption with Support for Boolean Queries,” *Advances in Cryptology - CRYPTO 2013*, Lecture Notes in Computer Science 3027, Springer-Verlag, 2013, pp. 353-373.

————, and Stefano Tessaro, “The Locality of Searchable Symmetric Encryption,” *Advances in Cryptology - EUROCRYPT 2014*, Lecture Notes in Computer Science 8441, 2014, pp. 351-368.

Chase, Melissa, and Emily Shen, “Substring-Searchable Symmetric Encryption,” *Proceedings on Privacy Enhancing Technologies*, Volume 2015, Issue 2, 2015, pp. 263-281.

Curtmola, Reza, Juan Garay, Seny Kamara, and Rafail Ostrovsky, “Searchable Symmetric Encryption: Improved Definitions and Efficient Constructions,” *Proceedings of the 13th ACM Conference on Computer and Communications Security*, 2006, pp. 79-88.

Faber, Sky, Stanislaw Jarecki, Hugo Krawczyk, Quan Nguyen, Marcel Rosu, and Michael Steiner, “Rich Queries on Encrypted Data: Beyond Exact Matches,” *Computer Security - ESORICS 2015*, Lecture Notes in Computer Science 9327, Springer-Verlag, 2015, pp. 123-145.

Gajek, Sebastian, “Dynamic Symmetric Searchable Encryption from Constrained Functional Encryption,” *Topics in Cryptology - CT-RSA 2016*, Lecture Notes in Computer Science 9610, Springer-Verlag, 2016, pp. 75-89.

Ishai, Yuval, Eyal Kushilevitz, Steve Lu, and Rafail Ostrovsky, “Private Large-Scale Database with Distributed Searchable Symmetric Encryption,” *Topics in Cryptology - CT-RSA 2016*, Lecture Notes in Computer Science 9610, Springer-Verlag, 2016, pp. 90-107.

Jarecki, Stanislaw, Charanjit Jutla, Hugo Krawczyk, Marcel Rosu, and Michael Steiner, “Outsourced Symmetric Private Information Retrieval,” *Proceedings of the 2013 ACM Special Interest Group on Security, Audit and Control Conference on Computer and Communications Cecurity*, 2013, pp. 875-888.

Kurosawa, Kaoru, “Garbled Searchable Symmetric Encryption,” *Financial Cryptography and Data Security*, Lecture Notes in Computer Science 8437, Springer-Verlag, 2014, pp. 234-251.

Li, Jin, Qian Wang, Cong Wang, Ning Cao, Kui Ren, and Wenjing Lou, “Fuzzy Keyword Search over Encrypted Data in Cloud Computing,” *Proceedings of IEEE Conference on Computer Communications*, 2010, pp. 441-445.

Popa, Ada Raluca, Catherine M.S. Redfield, Nickolai Zeldovich, and Hari Balakrishnan, “CryptDB: Protecting Confidentiality with Encrypted Query Processing,”

Proceedings of the 23rd ACM Symposium on Operating Systems Principles, 2011, pp. 85-100.

Yavuz, Attila A., and Jorge Guajardo, "Dynamic Searchable Symmetric Encryption with Minimal Leakage and Efficient Update on Commodity Hardware," *Selected Areas in Cryptography - SAC 2015*, Lecture Notes in Computer Science 9566, Springer-Verlag, 2015, pp. 241-259.

補論．検索可能暗号の各実現方式の処理性能評価の詳細

検索可能暗号の各実現方式にかかる処理性能の分析において、使用したパラメータを示す記号と設定値を図表 A-1 に示すとともに、評価結果の詳細を図表 A-2 に示す。

まず、評価尺度に影響を与えるパラメータについて説明する。既存の研究において、各評価尺度の値は、登録するファイルや登録キーワードの数等、さまざまなパラメータに依存する。そうした主なパラメータは、以下の 9 つである。すなわち、①登録するファイルの数（以下、「登録ファイル数」という）、②各ファイルに含まれる登録キーワードの数の平均値（以下、「登録キーワード数」という）、③登録キーワードの文字数の平均値（以下、「登録キーワード長」という）、④登録キーワードのバリエーションの数（以下、「全キーワード数」という）、⑤検索キーワードの文字数（以下、「検索キーワード長」という）、⑥検索キーワードの数（以下、「検索キーワード数」という）、⑦（一定のトラップドアに対して）正しい検索結果として得られるファイル（または、その識別子）の数（以下、「検索ヒット数」という）、⑧ファイルの暗号化等とは別に、乗算やべき乗の計算が必要な方式において、その計算の入力値（整数）のサイズ（以下、「整数サイズ」という）、⑨検索キーワードを構成する文字列に対して実施する文字の置換[26]の回数（以下、「置換回数」という）である。

図表 A-1　記号とパラメータの対応表

記号	パラメータ	評価に用いた値
N	登録ファイル数	変数
M	登録キーワード数	100 に設定
W	全キーワード数	変数
L	登録キーワード長	部分一致検索、類似検索、複数キーワード方式の比較では 12 に設定したほか、範囲検索の比較では 32 に設定
K	検索キーワード長	4 に設定
C	検索キーワード数	3 に設定
H	検索ヒット数	変数
q	整数サイズ	完全一致検索の比較では 2,048 に設定したほか、部分一致検索、複数キーワード方式の比較では 128 に設定
d	置換回数	2 に設定

[26] 文字の置換は、例えば、文字列「bank」（4 文字）のうち、「k」を「d」に置き換えて「band」とする処理であり、この場合、文字の置換の回数は 1 とする。文字の置換は類似検索で行われる。

図表 A-2　検索可能暗号の各実現方式の処理性能見積り

機能	方式	インデックス生成処理量（登録者）	インデックスのサイズ（登録者）	トラップドア生成処理量（登録者）	トラップドアのサイズ（登録者）	検索処理量（サーバ）	局所性	読取効率
完全一致検索	Curtmola *et al.* [2006]	O(MN)	O(MN)	O(N)	O(N)	O(N)	O(H)	O(1)
	Yavuz and Guajardo [2015]	O(MN)	O(MN)	O(1)	O(1)	O(N)	O(H)	O(1)
	Asharov *et al.* [2016]-1	O(MN)	O(MN)	O(1)	O(1)	O(log(MN))	O(1)	O(log (MN))
	Asharov *et al.* [2016]-2	O(MN)	O(MN)	O(1)	O(1)	O(log log(MN))	O(1)	O(log log (MN))
	Asharov *et al.* [2016]-3	O(MN)	O(MN log(MN))	O(1)	O(1)	O(1)	O(1)	O(1)
	黒澤ほか[2016]	O($q^3 + MN$)	O($N(M + q)$)	O(1)	O(1)	O(N)	O(H)	O(1)

機能	方式	インデックス生成処理量（登録者）	インデックスのサイズ（登録者）	トラップドア生成処理量（登録者）	トラップドアのサイズ（登録者）	検索処理量（サーバ）	局所性	読取効率
部分一致検索	早坂ほか[2016]	O(L^2MN)	O(L^2MN)	O(KW)	O(W)	O(L^2MN)	O(H)	O(1)
	平野ほか[2016]	O(L^2MN)	O(L^2MN)	O(1)	O(1)	O(L^2MN)	O(H)	O(1)
	Chase and Shen [2015]	O(LMN)	O(LMN log L)	O($H + K$)	O($H + K$)	O($H + K$)	O(H)	O(1)
	Faber *et al.* [2015]	O(L^2MNq^3)	O(L^2MNq)	O(Kq^3)	O(Kq)	O($N + HKq^3$)	O(H)	O(1)

機能	方式	インデックス生成処理量（登録者）	インデックスのサイズ（登録者）	トラップドア生成処理量（登録者）	トラップドアのサイズ（登録者）	検索処理量（サーバ）	局所性	読取効率
範囲検索	Popa *et al.* [2011]	O(MN)	O(MN)	O(1)	O(1)	O(N log N)	O(H)	O(1)
	Faber *et al.* [2015]	O(LMN)	O(LMN)	O(L)	O(L)	O(LN)	O(H)	O(1)

機能	方式	インデックス生成処理量（登録者）	インデックスのサイズ（登録者）	トラップドア生成処理量（登録者）	トラップドアのサイズ（登録者）	検索処理量（サーバ）	局所性	読取効率
類似検索	Li *et al.* [2010]	O(L^dMN)	O(L^dMN)	O(K^d)	O(K^d)	O(K^dN)	O(H)	O(1)
	Boldyreva and Chenette [2014]	O(MN)	O(MN)	O(N)	O(N)	O(HN log (MN))	O(H)	O(1)
	海上ほか[2016]	O(L^2MN)	O(LMN)	O(L^2M)	O(LM)	O(L^2MN)	O(H)	O(1)

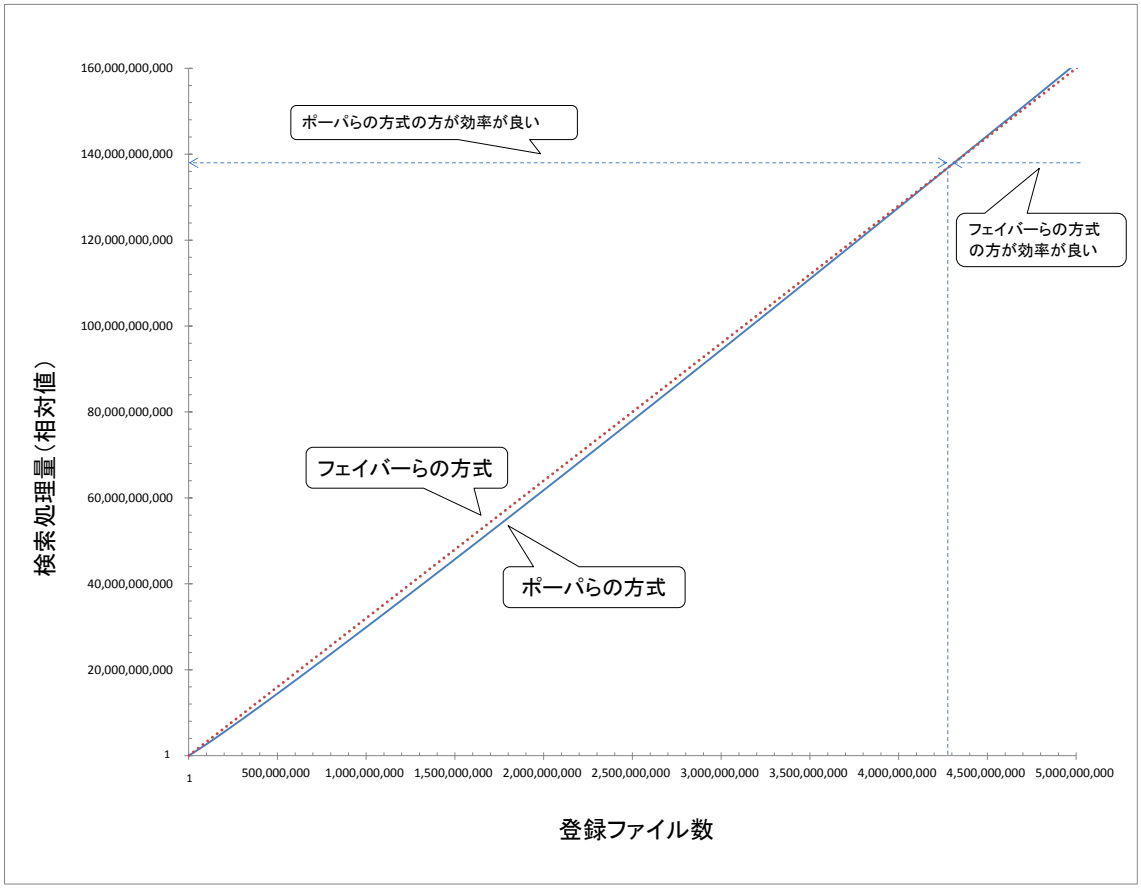
機能		方式	インデックス生成処理量（登録者）	インデックスのサイズ（登録者）	トラップドア生成処理量（登録者）	トラップドアのサイズ（登録者）	検索処理量（サーバ）	局所性	読取効率
複数キーワード検索	論理積検索	小嶋ほか[2016]	O(MN)	O(MN)	O(C)	O(M)	O(N log M)	O(H)	O(1)
	論理積と論理和の組合せ	Cash *et al.* [2013]	O(MNq^3)	O(MNq)	O(Cq^3)	O(Cq)	O($N + CHq^3$)	O(H)	O(1)
		Kurosawa [2014]	O(MN)	O(MN)	O(CN)	O($C + N$)	O(CN)	O(H)	O(1)
		Gajek [2016]	O(L^3MN)	O(LMN)	O(CL^3N)	O(CLM)	O(CL^3log M)	O(1)	O(1)

図表 A-2 で使用される記号「O」 は、各評価尺度が（O に続く）括弧内の関数の値の増加量に比例して増加することを表す。例えば、ある方式のインデックス生成処理量が、「O(登録ファイル数 × 登録キーワード数)」と評価されている場合は、その方式におけるインデックス生成処理量の増加量は、登録ファイル数と登録キーワード数の積の増加量に比例することを意味する。一般に、複雑な検索機能を実現する方式であるほど、インデックス生成処理量等の増加量が、登録ファイル数の増加量に比べて大きくなり、処理性能は低下することになる。したがって、不必要な検索機能を有する検索可能暗号を使用すると、必要な計算資源が過大となると考えられる。

図表 A-3 では、範囲検索の実現方式における検索処理量の評価・比較について、図表 7 よりも登録ファイル数をさらに大きくした際の値を示す。登録ファイル数が大きくなると（約 40 億以上）、相対的に望ましい方式が逆転する。

図表 A-4 では、複数キーワード方式の実現方式のうち、黒澤の方式とガジェックの方式の検索処理量の評価・比較について、図表 9 よりも登録ファイル数をさらに大きくした際の値を示す。登録ファイル数が大きくなると（約 1 万以上）、相対的に望ましい方式が逆転する。

図表 A-3　ポーパらの方式とフェイバーらの方式の検索処理量の比較



図表 A-4　黒澤の方式とガジェックの方式の検索処理量の比較