

メール演算システムにおける演算結果インタフェースの提案

近藤翔太[†] 齊藤義仰[†] 村山優子[†]

岩手県立大学ソフトウェア情報学部[†]

1. はじめに

複数人へのメール送信の場合、メーリングリストを用いることが多い。メーリングリストでは、あらかじめメールを受け取るメンバのメールアドレスを登録しておくことにより、メールを送るユーザはメーリングリストにメールを送信するだけで複数人へのメール送信が可能になる。課題として、一時的なメンバ変更が困難であるという点が挙げられる。

先行研究では、メーリングリストの宛先演算機能を備えたメールシステム Flexie¹⁾ を構築した。

宛先演算機能は、メーリングリストへの一時的なアドレスの追加、除外を行い、宛先を柔軟に変更することで、任意のメンバにメールを送信することが容易になった。しかし課題として、メール送信前に演算結果を確認する機能が無く、送信したいメンバへメールが送信されない、意図しないメンバへメールが送信されるなどの可能性が挙げられた。

本研究では、これらの課題を解決するため、先行研究で作成されたメールの誤送信を防ぐための Flexie 専用のインタフェースを改良し、演算結果の確認機能の実装した。さらに、誤送信を防止できたか検証するため、先行研究インタフェースとの誤送信回数の比較を行った。

2. 先行研究

本節では、先行研究 Flexie の機能と現状の課題から求められる機能について述べる。

Flexie は、メール毎にメーリングリストのメンバを変更することができるシステムである。従来のメール送信とは異なり、Flexie では宛先フィールドに Flexie が動作しているサーバのアドレス、コメント部に演算式を含んだメールアドレスを記述する。例えば、“ml@mailis.com” から kondo@mailis.com を除いた宛先を指定する場合は図 1 のように記述する。メールを受信した Flexie は、宛先の解析を行い、演算結果の宛先にメールを配送する。

Flexie では、上記のような独自のフォーマット

で宛先入力をするため、入力ミスが誘発されやすい。その手間を解消するため、先行研究では入力の簡易化のために Flexie 専用 Web インタフェースが作成された。Flexie 専用 Web インタフェースではメーリングリストの入力および演算子の入力、メール演算受け付け用アドレスの入力が不要になっている。しかし、メール送信前の確認機能がなく、メールが送信されない場合でもエラー通知がないため、送信したメンバへ送信されない可能性や意図しないメンバへのメール送信の可能性があった。

そこで、本研究では先行研究で実装された Flexie 専用 Web インタフェースに演算結果確認インタフェースを付加することによって、課題となった誤送信の可能性を防ぐ。

```
TO: "ml@mailis.com / kondo@mailis.com"
<flexie@mailis.com>
```

図 1 宛先指定の方法

3. 提案システム

本研究は、先行研究である Flexie の宛先演算結果の確認を促すことで誤送信を防ぐことを目的としている。システムモデルを図 2 に示す。

Flexie 専用 Web インタフェースを用いて、メールを送信するメーリングリスト、追加するメンバ、除外するメンバ、メール本文を入力しサーバへ入力したデータを送る。サーバ側で送られてきたデータをもとに、メールを送信するメーリングリストのメンバを抜き出し、演算の結果をインタフェースに表示する。演算結果を表示し、確認させて誤送信を防ぐ。その後、ユーザは演算結果を確認し適宜修正を行い最終的な送信を行う。そのメールのデータは Flexie へ送信され、演算を行った後、演算結果通りのメンバへメールが送信される。

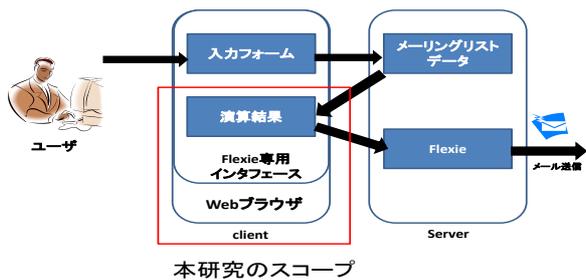
4. 実装

先行研究の段階で宛先入力の手間を省くために Flexie 専用 Web インタフェースが実装されていた。本システムは先行研究の課題である誤送信を防ぐため、Flexie 専用 Web インタフェースに付加する形で演算結果インタフェースの実装を行う。開発言語には Perl を使用した。これは、先行研究における既存のシステムが

An Proposal of an Operation Result Interface for the Flexible Mail Delivery System

Kondo Shota[†], Yoshia Saito[†], Yuko Murayama[†]

[†]Software and Information Science, Iwate Prefectural University



本研究のスコープ

図2 システムモデル

Perl で書かれているためである。演算結果インタフェースを図3に示す。

先行研究で実装されたFlexie専用Webインタフェースで送信するメーリングリストと追加するメンバもしくは除外するメンバを指定しFlexieへメールを送信する。その際、システム側で送信するメーリングリストのメンバを取り出しメンバの追加、除外を行い、インタフェースに表示する。確認ボタンはJavascriptで実装しており、この段階では更新ボタンと送信ボタンは押せなくなっている。更新ボタンを押した場合は、チェックボタンのチェックが付いているアドレスだけを判断し、メーリングリストのメンバとチェックの付いたアドレスを比較し、マッチしないアドレスが除外するメンバになる。除外するメンバを追加する場合は、チェックの付いたアドレスをそのまま追加する。

5. 評価

本インタフェースが誤送信を防止できるかどうかを検証するための評価を行った。評価方法は、先行研究と本インタフェースの2つを被験者が使用し、誤送信の回数を比較した。5~10名のメンバが登録された10のメーリングリストを用意し、メール送信を10回行った。

また、それぞれのメール送信にかかった時間を計測し比較した。被験者は岩手県立大学の学生15名で、誤送信とメール送信時間の平均で比較する。評価の最後に、被験者に対してアンケートに回答してもらった。

先行研究の平均誤送信回数が5回、本インタフェースの平均誤送信回数が0回であった。また、先行研究の平均メール送信時間が53.8秒、本インタフェースが59.1秒であった。これらの結果から、演算結果を確認させることは誤送信防止に有効であり、確認時間がかからないことが分かった。これらの結果から、先行研究Flexieにおける誤送信の防止については、演算結果を確認させることが有効であると考えられる。また、先行研究のインタフェースの改善、



図3 演算結果インタフェース

本インタフェースの見易さの改善、アドレスを打ち間違えた場合の強調機能の実装などが必要であることが分かった。

6. 関連研究

CipherCraft/Mail²⁾はメール送信における宛先確認機能を追加することにより、メール誤送信機能を実装したメーラーである。ユーザが送信ボタンを押した時、送信確認ダイアログを表示し、宛先や添付ファイルの再チェックを行うことができる。項目全てのチェックボックスにチェックを入れることで送信が行われる。また、それらの要素を総合的に判断し、送信しようとしているメールの安全性を、点数と3段階のアイコンを用いて通知を行い、誤送信を防止する。当該研究が通常のメール送信における宛先確認機能であることに対し、本研究では、先行研究Flexieによる演算結果の確認機能を実装した。

7. おわりに

本研究では、先行研究における課題であった誤送信を防ぐため、演算結果を確認し、メール演算が正常に行われるかチェックを行う演算結果確認インタフェースの実装を行った。本研究の課題としては、アドレス打ち間違いなどを起こした場合の強調機能がないため、ユーザが間違いに気づかない可能性がある。今後、解決策として、不快なインタフェース³⁾を本研究に適用すれば、間違いの強調が可能になろう。

参考文献

- 1) 加藤順也：柔軟なメール配送を実現するFlexieの実装と評価、岩手県立大学ソフトウェア情報学部 卒業論文 (2006)。
- 2) CipherCraft/Mail, <http://www.ntts.co.jp/products/ccraftmail/index.html>
- 3) 藤原康宏, 村山優子：コンピュータ利用時の不快感を利用した警告インタフェースの提案, 情報処理学会誌, Vol152, No. 1. pp. 77-89 (2011)。