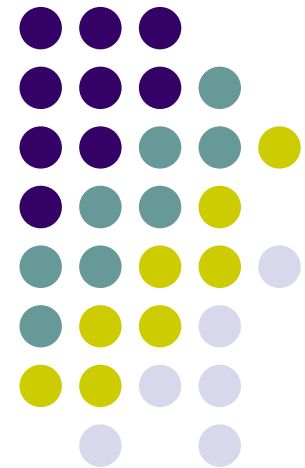


どこでテストをやめるのか？

2008.1.30

日本電気株式会社

川村真弥





本日の講演内容

- テストのやめ時はいつか？
- 信頼度成長曲線
- 収束判定
- テストの十分性
- まとめ

出典「ソフトウェア・テストPRESS Vol.1
『どこでテストをやめるか？』
伊藤拓也(NEC)著」



テストのやめ時はいつか？



テストの終了基準

現実的によくあるパターン

- 決められた納期がきたので
- 決められた工数に達したので
- 計画したテスト項目を全て実施したので
- 計画した数のバグを摘出したので

意識の高い人の疑問

- テストをもっとすれば、まだバグが増えるのでは？
- ここでテストをやめることで、どの程度のリスクがあるのか？

→ これらの問いに答えるにはどうしたらよいだろうか？

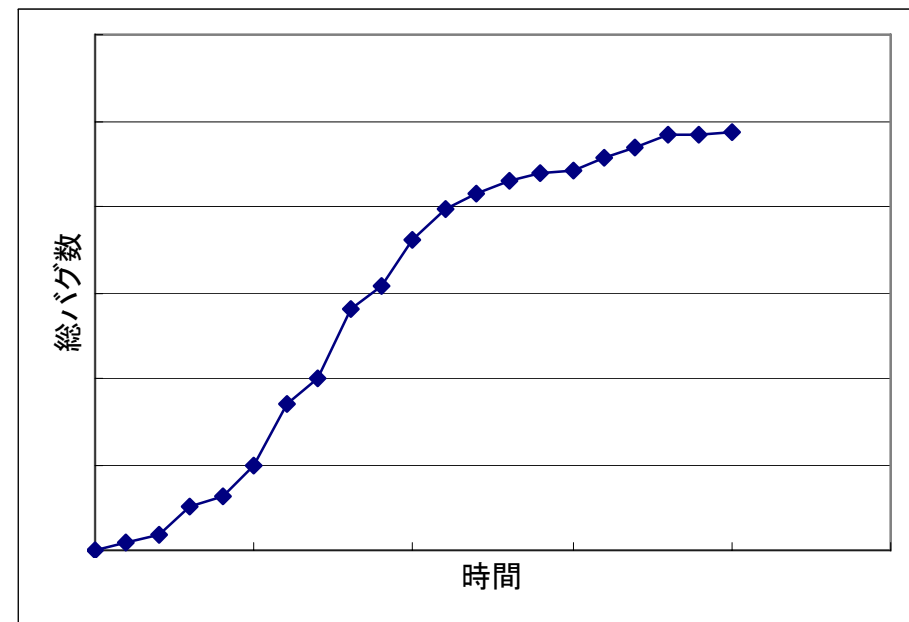


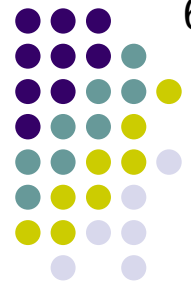
信頼度成長曲線と収束判定(1)

テスト工程でバグが収束すると、出荷後にお客様の環境でバグが出にくいことが経験的に知られている

バグが収束したかどうかで以下の点を確認可能

- これ以上テストをしてもバグが出そうにない
- 出荷後にバグが発見されるリスクが一定の水準以下である

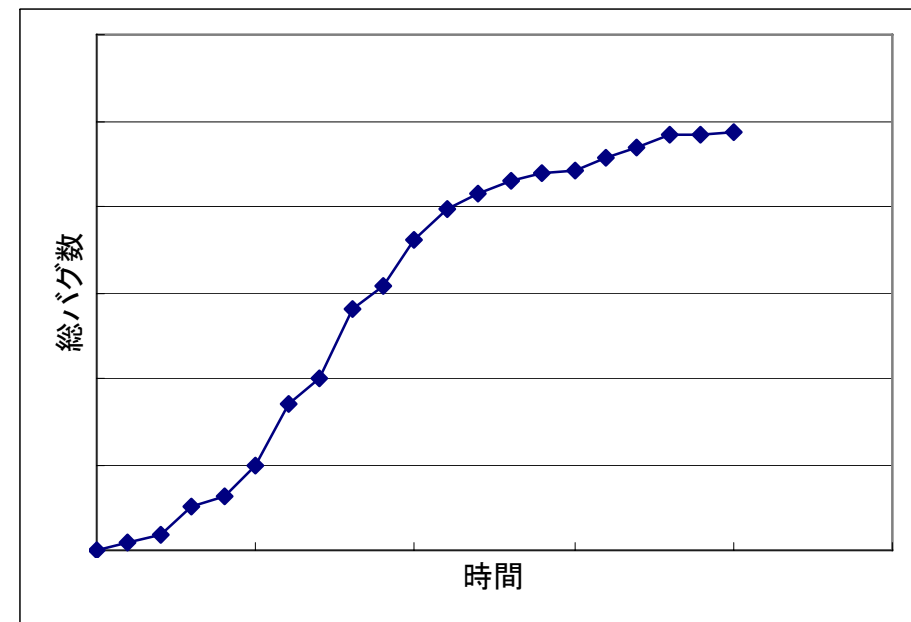




信頼度成長曲線と収束判定(2)

バグが収束したかどうかを判断するためには、一般的に信頼度成長曲線を使う

- 信頼度成長曲線は使い方を誤ると収束の判断を誤ることになる
- 具体的にどのように収束を判断するのかよくわからない



→ 実際に使いこなすのは意外と難しい

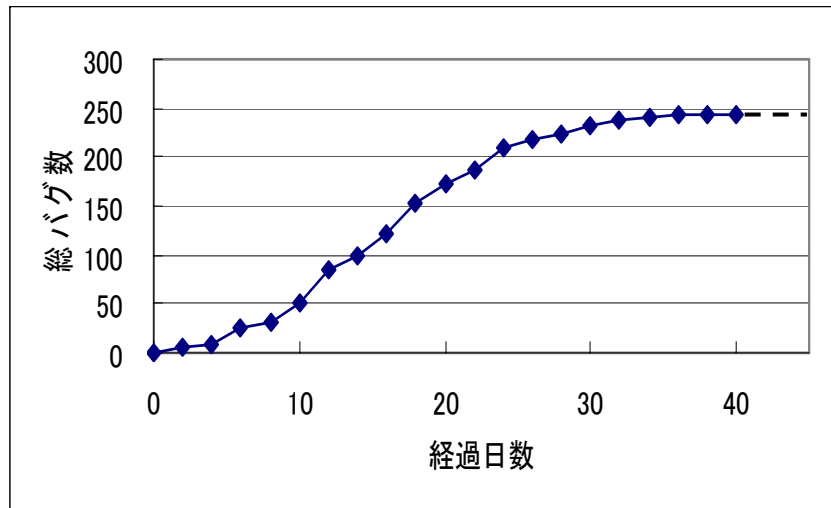


信賴度成長曲線



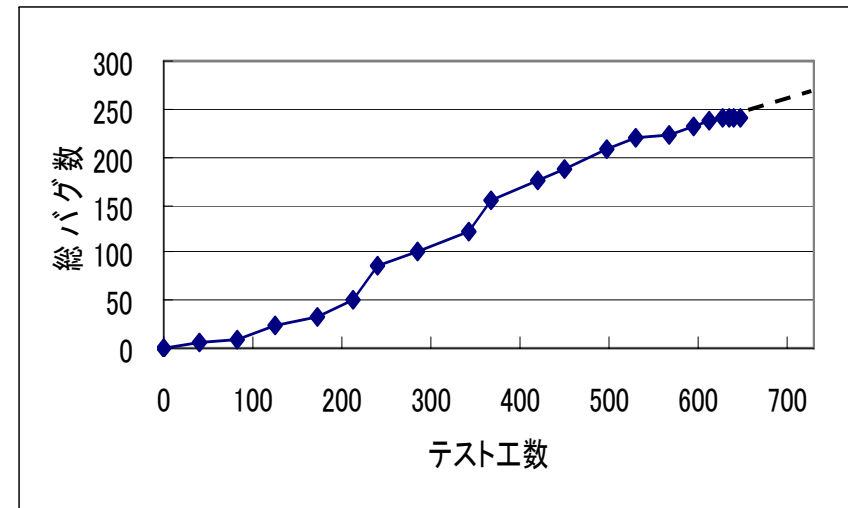
信頼度成長曲線の落とし穴

横軸に何を取るかで見え方が大きく変わってくる



横軸に「経過日数」を取った場合

→ 収束している



横軸に「テスト工数」を取った場合

→ 収束していない

なぜこのようなことが起こるのか？



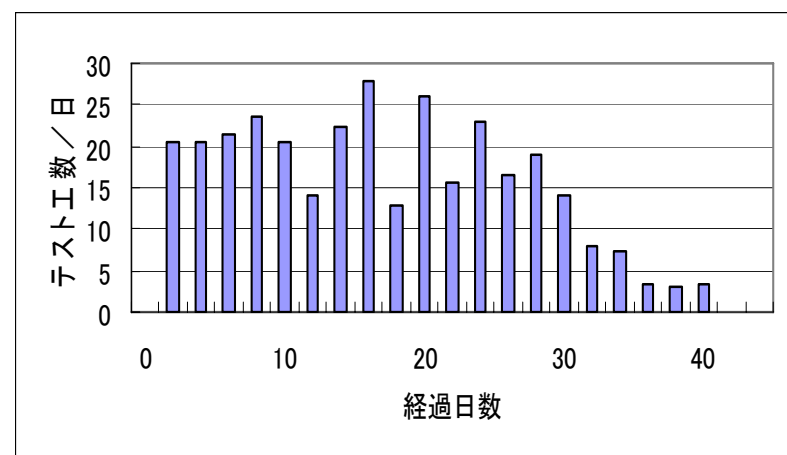
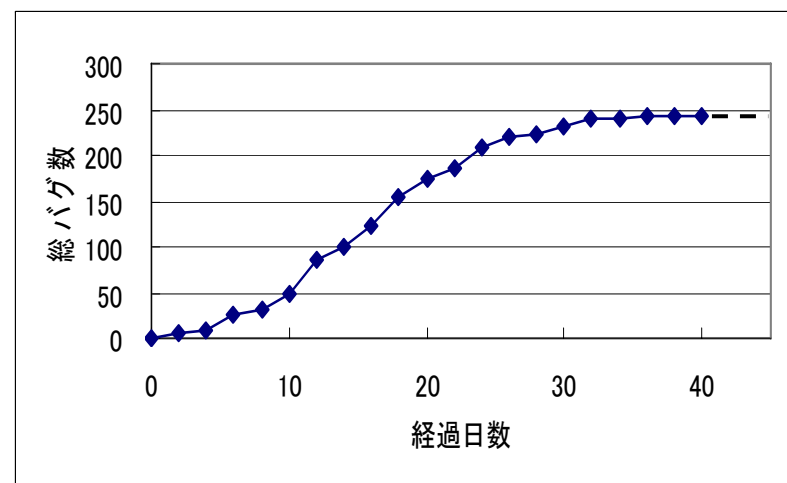
種明かし

答え：テストの終盤で、日ごとの
消化項目数が減ってきたから

(この理屈なら、テストをやめれば
いずれ必ず収束する)

→ 横軸に「**経過日数**」を取るのは
当てにならない

横軸には、
「**テスト工数**」か
「**テスト項目数**」
をとった方がよい





テストの時間的推移とバグ数の推移

テストの時間的推移に従い、テストの内容は変わっていく
テストの内容が変われば、新しい内容のバグが抽出され、
抽出バグ数は一時的に増える

(例)テストの時間的推移による、テスト内容・抽出バグ内容の変化

時間的推移	テスト内容	抽出バグ内容
序盤	単機能・正常系・代表的異常系	単純バグ
中盤	異常系・複数機能・バリエーション	タイミング・複合条件バグ
終盤	大規模・特殊環境・性能	諸元依存・環境依存・性能問題

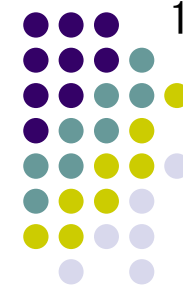
テストの進め方は組織によって千差万別

→ バグ数の推移の特徴も組織によって違う

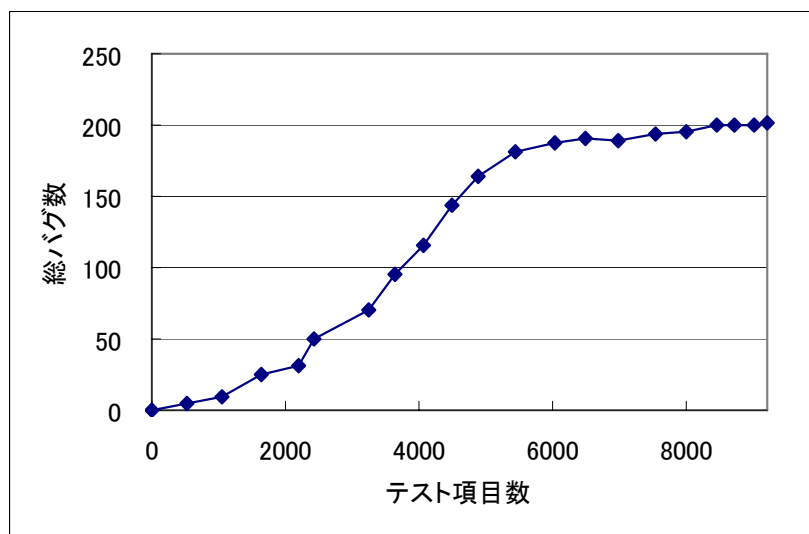
信頼度成長曲線の種類は自分の組織に合ったものを選ぶ



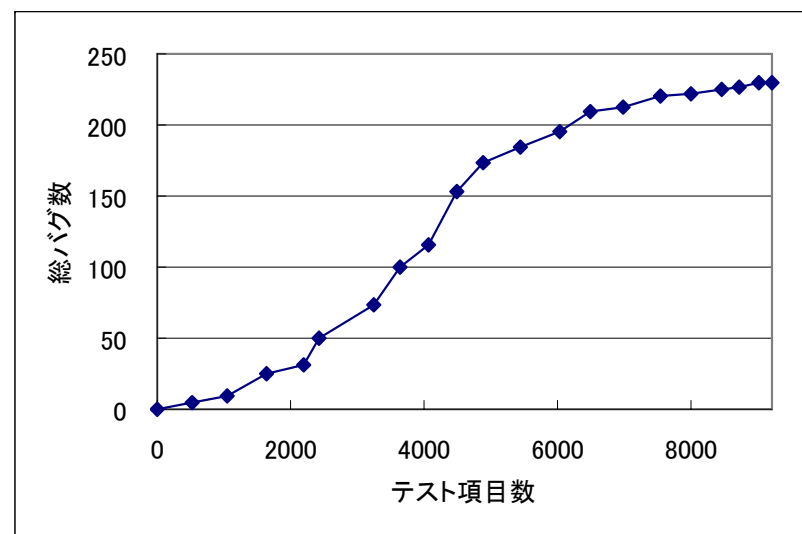
収束判定



収束判定はどのようにすべきか？(1)



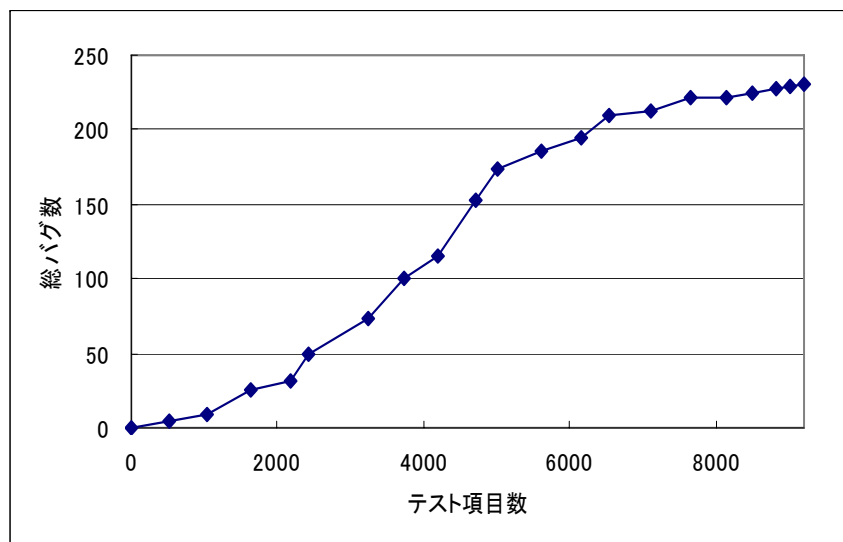
「収束している」で問題なさそう



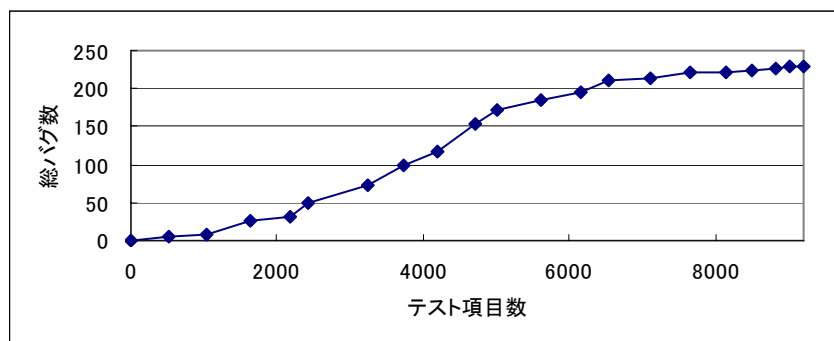
微妙・・・人によって意見が違いそう



収束判定はどのようにすべきか？(2)



一見収束していないようにも・・・



スケールを変えれば見え方も変わる

誰が判断しても同じ結果が得られる方法が必要



ツールの使用

長所

- 収束値が明確

問題点

- プロジェクト規模が小さい(=総バグ数が少ない)場合に、バグ摘出推移グラフが大きな階段状になる
- テスト工程の紆余曲折により、信頼度成長曲線とはかけ離れたグラフになる
- 複数種類の信頼度成長曲線が選択できるが、曲線によって結果が異なるため、選択を迷う



傾きを利用した収束判定

「テスト開始から終了までの全体の傾き」に対する、「テスト終盤の傾き」を比較する

- 信頼度成長曲線の種類によらない(どの曲線でも終盤は平坦)
- 機械的に収束判定できる

$$\text{収束度} = \frac{\text{テスト終盤の傾き}}{\text{テスト開始から終了までの全体の傾き}}$$

収束の基準値を設けて、その基準値以下であれば収束していると判定する

基準値 \geq 収束度 \rightarrow 収束している

基準値 $<$ 収束度 \rightarrow 収束していない

収束の基準値については、各組織の品質実績を元に設定する

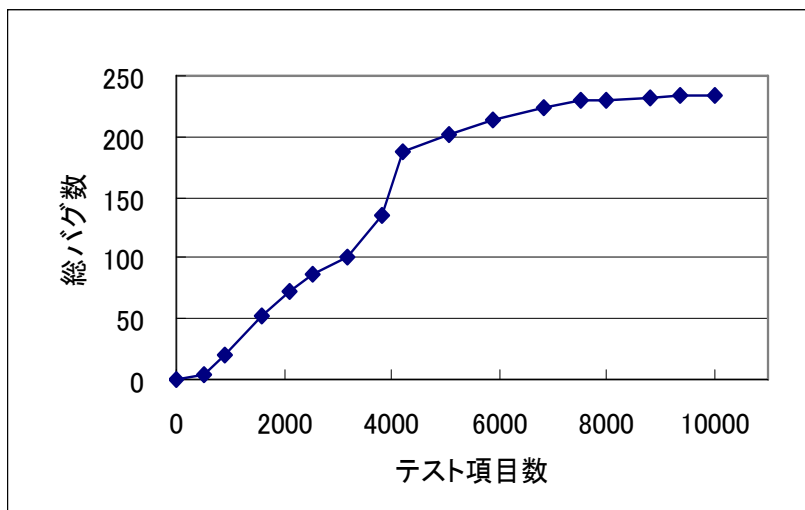


収束の基準値設定のポイント

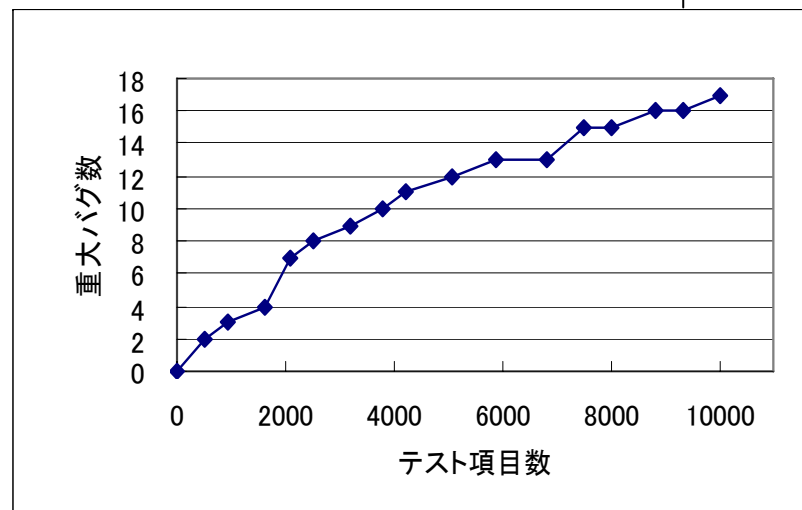
- 収束度に対する出荷後バグ発生件数の実績値などから、品質目標を達成するために必要な基準値を決定する
- 終盤の定義（テスト消化率80%～100%、90%～100%など）について明確に決める
- テストの進捗段階ごとに、収束の基準値を下げていってもよい



重大バグを考慮した収束判定



ソフトウェアシステム全体(収束)



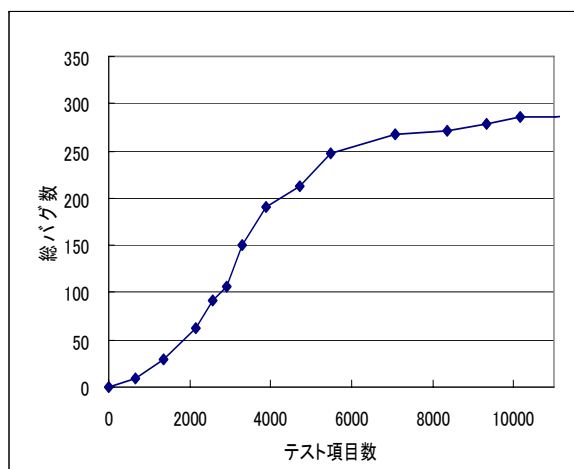
重大バグ(収束していない)

重大バグの収束を考慮することで、出荷後にバグが発生した時の対応リスクを減らすことができる

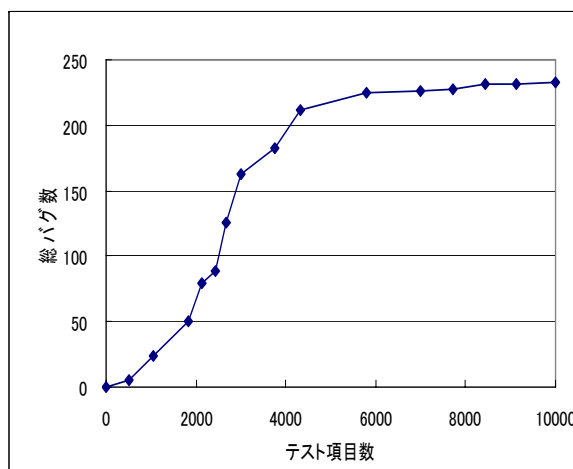
重大バグの定義は明確にしておくことが重要



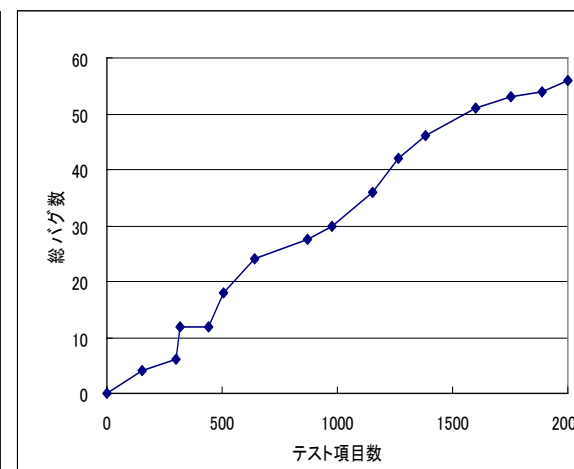
機能毎の収束判定



ソフトウェアシステム全体(収束)



機能A(収束)



機能B(収束していない)

- 小規模の(=総バグ数が少ない)機能の品質の確認
- 特定機能に偏ったバグの検出

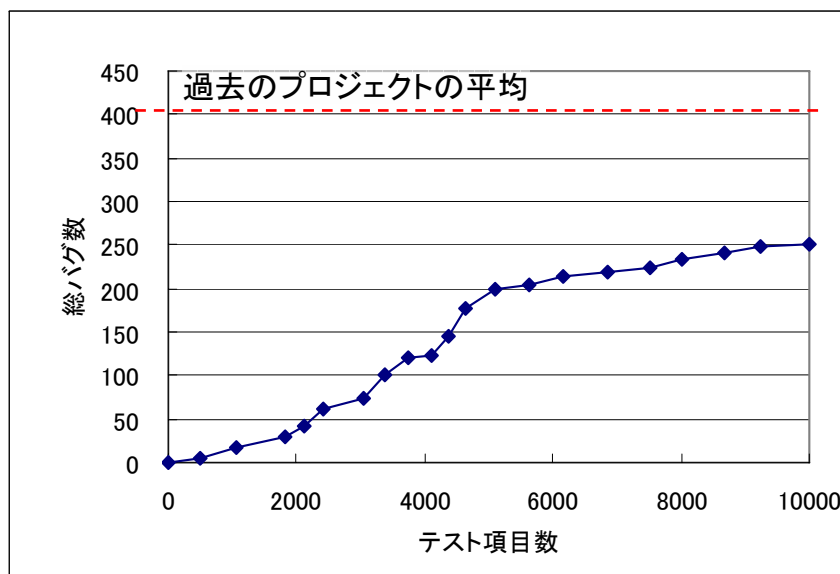
追加テストによる弱点機能の品質強化が必要



テストの十分性



過去データの利用による検証



収束しているように見えるが、バグの抽出率が過去のプロジェクトの平均よりかなり低い

妥当な理由がある

(開発者レベル・開発難易度の差等)

→ 問題なし

妥当な理由がない

→ 追加テストが必要

(テストの進め方の問題・

テスト観点不足の可能性)

過去データとの差分の理由を検証することで、収束観点に隠れたテストの十分性を確認



追加テスト

追加テストが必要な場合

- 予定したテスト項目数でバグが収束しない場合
- 過去データとの差分について妥当な理由がない場合
- バグの傾向が偏っている場合（発生原因・発生箇所）

追加テストの観点

- バグが抽出された機能
- バグと類似した実装・設計をした箇所
- バグ修正による影響範囲の再確認

追加テストの質が最終的な品質を左右する



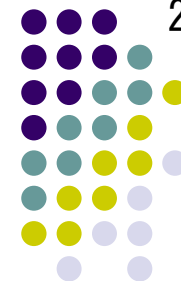
まとめ



まとめ

テストのやめ時は、信頼度成長曲線が収束した時である

- 信頼度成長曲線
 - 横軸には「テスト工数」か「テスト項目数」を
- 収束判定
 - 信頼度成長曲線の種類は問わない
 - 傾き(収束度)を利用した判定
 - 重大バグを考慮した判定
 - 機能毎の判定
- テストの十分性
 - 過去データの利用による検証
 - 追加テストの観点の吟味



御清聴ありがとうございました