数理解析研究所講究録
第1636巻 2009年 251-258

アジャイル・ソフトウェア開発における定量的ソフトウェア品質評価法に関する考察

鳥取大学大学院・工学研究科　青木 俊樹 (Toshiki Aoki)[†]
鳥取大学大学院・工学研究科　山田　茂 (Shigeru Yamada)[†]
[†]Graduate School of Engineering, Tottori University

1　まえがき

近年，消費の多様化に伴って，サービスおよび製品のライフサイクルは短縮化する傾向にあり，その影響は製品に付随する情報システムや組込みソフトウェア開発の低コスト化・短納期化にも及んでいる [1]．さらに，ソフトウェア開発の現場では，商品競争力を高めるために顧客の要求仕様が競合の動向に合わせて変化したり，実現するべき機能が明確に決まっていないのにもかかわらず開発に着手し始めなければならない状況が頻繁に生じている．そこで近年では，従来の品質を保ちながら短納期・低コスト・仕様変動といったような三重苦を克服するため，迅速かつ適応的にソフトウェアを開発することができるアジャイル・ソフトウェア開発が注目されており，小・中規模なアプリケーション開発や，顧客の要件定義や要望が曖昧である場合，継続的なリリースが必要になる社内システムなど，仕様の変更が頻繁に行われる開発に多く適用されている．

アジャイル・ソフトウェア開発の主な目的は，顧客に対して迅速に価値を提供すること，変化に素早く対応することなどが挙げられ，近年のソフトウェア開発においては必要不可欠な開発手法となっている．しかしながら，アジャイル・ソフトウェア開発のプロセス計測データを採取するのは非常に困難であり，開発プロジェクトの的確な定量的評価法は確立されていない．したがって，アジャイル・ソフトウェア開発手法を用いたソフトウェア開発の現場では，経験則・暗黙知により品質・信頼性を判断することが多く，定量的な評価を実施することなくリリースされているのが現状となっている．

そこで本論文では，実際にP社において収集されたアジャイル・ソフトウェア開発におけるプロセス計測データを用いて，品質・信頼性の観点から開発プロジェクトの定量的な評価を行う．まず，重回帰分析 [2] を適用することにより，導出されたソフトウェア品質予測モデルからソフトウェア製品品質に影響を及ぼす要因を明らかにする．次に，品質・信頼性に関係があると推測される幾つかのメトリクスを用いて，ソフトウェア信頼性評価 [3] を実施し，アジャイル・ソフトウェア開発におけるソフトウェア信頼性評価法の有用性を考察する．

2　分析データ

本論文では，アジャイル・ソフトウェア開発におけるプロセス計測データを対象として分析を行う．アジャイル・ソフトウェア開発とは，ソフトウェア工学の考え方に基づいて，迅速かつ適応的にソフトウェアを開発する軽量な開発手法の総称を表し，従来型の開発手法であるウォーターフォールモデルなどの計画駆動型開発手法と対極に位置する開発手法として知られている．

アジャイル・ソフトウェア開発では，プログラムを常に実行可能な状態に保ち，確認・拡充していくインクリメンタル手法（段階的拡充手法）をとる．開発対象を多数の小さな機能に分割し，1つのイテレーション（反復）で1機能を開発する．このイテレーションのサイクルを繰り返し行い，それまでに開発した成果物に機能を1つずつ追加していく．このようにすることによって，仕様変動のリスク，技術的実現性のリスクを回避しながら，各イテレーションが終了する毎に機能が追加された新しいソフトウェアをリリースすることを目指す．1つのイテレーション内では，要求定義，設計，コーディング，テストといったソフトウェア開発プロジェクトに要する一連の開発プロセスが実行され，各イテレーションに要する期間は，数日から数週間と短いのが通例である．

ここで，1つのイテレーション内に含まれる各工程における主な特徴を以下に示す．まず，要求定義においては，顧客から得られた要求をもとに開発対象を1ヶ月未満で開発可能なサイズに分割し，顧客にとって

表 1： 相関分析表

	X_1	X_2	X_3	X_4	X_5	Y
X_1	1					
X_2	0.853	1				
X_3	0.581	0.742	1			
X_4	0.727	0.836	0.976	1		
X_5	0.264	0.663	0.865	0.800	1	
Y	0.538	0.529	-0.166	0.021	-0.131	1

重要性の高い機能，仕様が確定している機能から優先的に開発に取りかかる．設計およびコーディングにおいては，シンプルな設計・コーディングを行うことで無駄な作業を極力削減することを目指し，従来型開発手法の問題点に対処するために，ペア・プログラミング（一人がコードを書き，もう一人がそれをチェックしながらナビゲートする）という方法を適用している．テストにおいては，実装を行うよりも先に，実行可能なテストケースを作成し，実装される機能を明確にすることで，シンプルな設計を可能にし，フォールト修正による手戻り工数を極力少なくする方法をとる．以上のように，開発工程において極力無駄を省くことで低コスト・短納期での開発を図る．

本論文では，実際にアジャイル・ソフトウェア開発を適用したプロジェクトデータを分析に使用する．各プロジェクトで採取可能なメトリクスは，結合数，障害件数，レビュー回数，テストケース数，開発規模，開発工数，およびST 検出フォールト件数であり，イテレーション毎に計測されたものである．

3 重回帰分析

本論文では，取り扱うプロセス計測データが多変量であるため，多変量解析の 1 つである重回帰分析を用いる．重回帰分析とは，目的変量が説明変量の変動によってどの程度影響されるかを分析し，説明変量から目的変量を線形式により推定・予測する方法である．その関係式から，結果に大きな影響を与えている要因を明らかにすることができる．重回帰分析においては，各プロジェクトのイテレーションごとに採取されるメトリクスの値を統合したものを変量として取り扱う．

3.1 相関分析

説明変量として扱う 5 つのメトリクス $X_i(i$=1,2,3,4,5) と目的変量 (Y) の相関分析を行うと表 1 の結果が得られ，下記の相関関係が考えられる．

- 開発規模当り結合数 (X_1) および開発規模当り障害件数 (X_2) は，開発規模当り ST 検出フォールト数 (Y) との相関が高い．
- 開発規模当り ST 検出フォールト数 (Y) は，開発規模当り開発工数 (X_4) との相関が非常に低い．
- 開発規模当り結合数 (X_1) と開発規模当り障害件数 (X_2)，開発規模当りテストケース数 (X_3) と開発規模当りレビュー回数 (X_5) との間の相関は高く，多重共線性の可能性がある．

以上より，開発規模当り結合数 (X_1) と開発規模当り障害件数 (X_2)，開発規模当りテストケース数 (X_3) と開発規模当りレビュー回数 (X_5) との間に多重共線性がある可能性を考慮し，分散比の変動，目的変量との相関の高さを総合的に評価した結果，開発規模当り障害件数 (X_2)，開発規模当りテストケース数 (X_3) を重回帰分析の説明変量として用いる．

3.2 分散分析

重回帰分析における回帰精度および分散分析表は，それぞれ表 2 および表 3 になる．表 2 の回帰精度より，補正決定係数 R^2 は 0.943 という非常に高い値となる．また，表 3 の分散分析表より，

$$F_0 = 34.115 > F_2^2(0.05) = 19.00,$$

表 2： 回帰精度

重相関係数 R	0.986
決定係数 R^2	0.972
補正決定係数 R^2	0.943
標準誤差	0.401

表 3： 分散分析表

要因	自由度	変動	分散	検定統計量 Fo
回帰	2	10.951	5.475	34.115*
残差	2	0.321	0.160	
計	4	11.272		

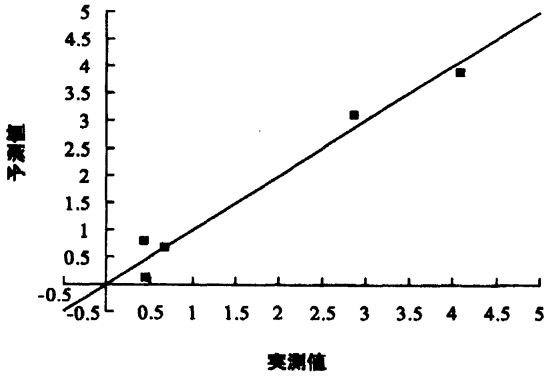

図 1： 開発規模当り ST 検出フォールト数 (Y) の予測精度

となり，危険率 5 %で有意となり，得られた重回帰式が予測に役立たないという帰無仮説は棄却される．以上より，重回帰式のデータに対する適合性は高いといえる．

3.3 推定された重回帰式

重回帰分析より，式 (1) の重回帰式 $\hat{Y}$ が導出される．また，分析するデータを標準化して回帰分析を行った結果，式 (2) の標準化重回帰式 $\hat{Y}^N$ が導出される．

$$\hat{Y} = 1.574 \cdot X_2 - 0.012 \cdot X_3 - 1.553, \quad (1)$$

$$\hat{Y}^N = 1.448 \cdot X_2 - 1.240 \cdot X_3. \quad (2)$$

式 (2) より，標準偏回帰係数の絶対値を比べると，説明変量の目的変量に影響を与える度合の大きさは，$X_2 > X_3$ であることがわかる．また，開発規模当り障害件数 (X_2) と開発規模当りテストケース数 (X_3) が目的変量である開発規模当り ST 検出フォールト数 (Y) に大きな影響を与えているといえる．

3.4 重回帰式によるソフトウェア品質の予測

本論文では，重回帰分析の精度を向上させるため，開発規模当りに規準化したデータを用いて分析を行った．式 (1) にアジャイル開発下におけるプロセスデータを代入した予測値と，実測値である開発規模当り ST 検出フォールト数 (Y) との関係を図 1 に示す．図 1 より，実測値と予測値はほとんど誤差がないことがわかる．よって，開発規模当り ST 検出フォールト数 (Y) は非常に高い精度で予測でき，データに対する適合性は高く，プロセスデータを開発規模当りに規準化したことが精度の向上に大きく寄与したと考えられる [4].

3.5 重回帰分析からのプロジェクト評価

- 開発規模当り障害件数 (X_2) が開発規模当り ST 検出フォールト数 (Y) に最も大きな影響を与えていることから，モジュールを結合する前の段階でモジュール内の障害件数を抑えることが重要であることがわかる．

- 開発規模当りテストケース数も開発規模当り ST 検出フォールト数 (Y) に影響を与えており，テストケース数を増やせば開発規模当り ST 検出フォールト数 (Y) を減らすことができるが，納期・コストを考慮し，適切かつ適量のテストケース数を設定することが重要であることがわかる．

4 ソフトウェア信頼性評価

ソフトウェア品質の計測方法として，ソフトウェア信頼性評価技術がある．その中でも，ソフトウェア信頼度成長モデル（Software Reliability Growth Model，以下 SRGM と略す）は，動的環境におけるソフトウェアの挙動を信頼度成長過程として記述するものであり，開発中のソフトウェアに含まれる不具合数やソフトウェア信頼度を推定する方法としてよく知られている．このモデルは実際の適用例も多く，ソフトウェア信頼性モデルの中でも中心的役割を担っている．アジャイル・ソフトウェア開発におけるソフトウェア信頼度成長曲線は，イテレーション回数とイテレーション開発終了後の統合システムによるテストにおいて発見されたフォールトの累積数との関係を示す．

本論文では，アジャイル・ソフトウェア開発におけるプロセス計測データに非同次ポアソン過程（nonhomogeneous Poisson process，以下 NHPP と略す）に基づいた SRGM を適用することにより，発見されるフォールトの挙動を捉え，ソフトウェア信頼性の定量的評価を行う [5]．NHPP モデルは，適用性の観点から有望視され，多くの企業でも実用されているモデルの一つである．

4.1 離散化 NHPP モデルの適用

アジャイル・ソフトウェア開発の特徴より，イテレーション毎に採取されるメトリクスを従来の信頼性評価におけるテスト時間の代替メトリクスとする．したがって，テスト時間の代替メトリクスとして用いる値は離散値となるため，差分方程式に基づいて導出された離散化 NHPP モデルを信頼性評価に適用する．離散化 NHPP モデルの適用においては，テスト時間の代替メトリクスとして差分間隔が一定であるイテレーション数を使用する．イテレーション j までに発見される ST 検出フォールト数の総数を表す計数過程 $\{N_j, j \geq 0\}(j = 0, 1, 2 \cdots)$ が平均値関数 H_j をもつ離散化 NHPP に従うものと仮定すると，SRGM は

$$\Pr\{N_j = n\} = \frac{\{H_j\}^n}{n!} \exp[-H_j] \qquad (n = 0, 1, 2, \cdots), \tag{3}$$

$$H_j = \sum_{x=0}^{j} h(x), \tag{4}$$

と表現できる．

本論文では，基本的過程が微分方程式で表される従来の NHPP モデルの大域的性質（厳密解の存在）を保存するように導出された離散化指数形 SRGM（以下 DEXP と略す）[6] および離散化習熟 S 字形 SRGM（以下 DIS と略す）[6] を用いて，最小二乗法によりパラメータを推定し，信頼性評価を行う．

4.2 連続型 NHPP モデルの適用

次に，解析的取り扱いが比較的容易な連続時間を仮定した SRGM を用いて信頼性評価を行う．イテレーション j までに発見される ST 検出フォールト数の総数を表す計数過程 $\{N(j), j \geq 0\}$ が平均値関数 $H(j)$ をもつ NHPP に従うものと仮定すると，SRGM は

$$\Pr\{N(j) = n\} = \frac{\{H(j)\}^n}{n!} \exp[-H(j)] \qquad (n = 0, 1, 2, \cdots), \tag{5}$$

$$H(j) = \int_0^j h(x)dx, \tag{6}$$

となる．式 (5) において $H(j)$ は $N(j)$ の期待値であり，イテレーション j までに発見される総期待フォールト数を表す．アジャイル・ソフトウェア開発の特徴から，イテレーションは分析メトリクスの重要な要素として捉えることができる．したがって，各イテレーションにおいて採取可能なイテレーション回数，結合数，障害件数，レビュー回数，テストケース数，開発規模（LOC）および開発工数（人日）を離散的データとして捉え，式 (5) および式 (6) における j にイテレーション毎に採取されたメトリクスの累積値を代入し，テスト時間の代替メトリクスとすることにより信頼性評価を実施する．

まず，最終となるインクリメントを結合した後のシステムテスト完了時を，従来の信頼性評価におけるテスト終了時期と仮定し，ST 検出可能フォールト数が有限であると考え，信頼性評価に用いるモデルを指数

表 4： MSE に基づく適合性評価結果

Data Set	SRGM	イテレーション数	結合数	障害件数	レビュー回数	テストケース数	開発規模	開発工数
ProjectA	EXP	6.75	7.58	7.28	2.97	7.82	-	2.72
	DSS	10.56	11.43	5.54	2.32	6.42	**1.58**	4.62
	LPE	6.42	7.50	7.72	3.10	-	-	-
	DEXP	**2.08**						
	DIS	2.16						
ProjectB	EXP	6.47	-	-	3.85	2.66	2.17	3.25
	DSS	3.94	1.48	3.96	2.29	1.96	**1.26**	1.90
	LPE	6.44	-	-	5.71	-	2.15	34.12
	DEXP	**2.06**						
	DIS	3.12						
ProjectC	EXP	-	60.97	46.59	-	21.97	33.25	44.88
	DSS	22.85	32.27	24.26	39.27	**5.26**	10.52	23.05
	LPE	-	62.68	48.20	-	-	-	-
	DEXP	5.40						
	DIS	**4.95**						
ProjectD	EXP	-	-	-	-	39.25	-	-
	DSS	**9.15**	10.00	28.12	16.63	28.20	28.68	10.03
	LPE	-	-	-	-	40.34	-	-
	DEXP	-						
	DIS	**2.83**						

表 5： AIC に基づく適合性評価結果

		イテレーション数	結合数	障害件数	レビュー回数	テストケース数	開発規模	開発工数
ProjectA	EXP	38.00	45.18	36.03	30.39	39.84	-	30.11
	DSS	40.40	45.50	33.73	30.13	36.50	**29.36**	32.04
	LPE	37.86	45.16	36.09	30.41	-	-	-
ProjectB	EXP	35.41	-	-	31.08	28.49	26.37	29.50
	DSS	31.56	24.15	28.91	27.79	25.59	**23.54**	26.52
	LPE	35.43	-	-	31.12	-	26.69	29.55
ProjectC	EXP	-	75.17	60.02	-	41.26	52.86	66.80
	DSS	48.54	57.54	45.30	64.64	**29.36**	38.07	52.18
	LPE	-	75.96	60.13	-	-	-	-
ProjectD	EXP	-	-	-	-	74.08	-	-
	DSS	44.96	47.70	96.66	59.20	66.97	65.05	**44.18**
	LPE	-	-	-	-	74.18	-	-

形 SRGM（以下 EXP と略す）[7]，遅延 S 字形 SRGM（以下 DSS と略す）[7] とする．次に，アジャイル・ソフトウェア開発は段階的に拡充していくインクリメンタル手法を用いるため，最終的に検出される ST 検出可能フォールト数は無限となると考えることもできる．よって，検出可能フォールト数が無限である場合を仮定した対数型ポアソン実行時間モデル（以下 LPE と略す）[7] を信頼性評価に用いるモデルに加える．以上の 3 つの連続型 NHPP モデルを信頼性評価に使用し，モデルに対する適合性評価を行う．

4.3 適合性評価

本論文では，適合性を比較するための評価基準として，平均偏差平方和（mean squared error，以下 MSE と略す）および赤池情報量基準（Akaike information criterion，以下 AIC と略す）を使用する [8]．MSE はフォールト発見数データと推定値の誤差を直接比較するものであり，n 回のイテレーションが観測された場合，MSE は次式で表される．

$$\mathrm{MSE} = \frac{1}{n}\sum_{k=1}^{n}[y_k - \hat{H}(j_k)]^2. \tag{7}$$

AIC は自由パラメータ数が異なる SRGM の良し悪しを比較するために，フォールト発見数データに対する適合性の良さとモデルの単純さの兼ね合いで最適モデルを評価する基準であり，次式によって表される．

$$\mathrm{AIC} = 2 \times (M - \mathrm{MLE}). \tag{8}$$

ここで，M および MLE は SRGM における自由パラメータ数およびモデルの最大対数尤度を表す．比較する SRGM の AIC の値の差が 1 以上ある場合，小さい値をもつ SRGM が良いモデルであると判断できる．

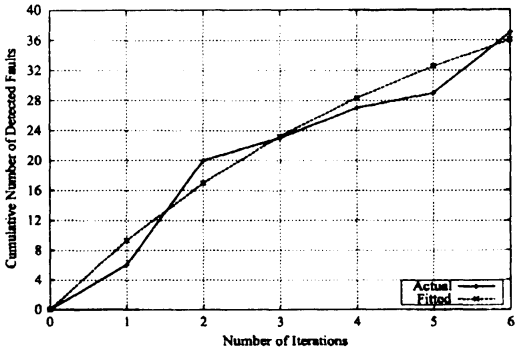

図 2： 推定された平均値関数 $\hat{H}_n$

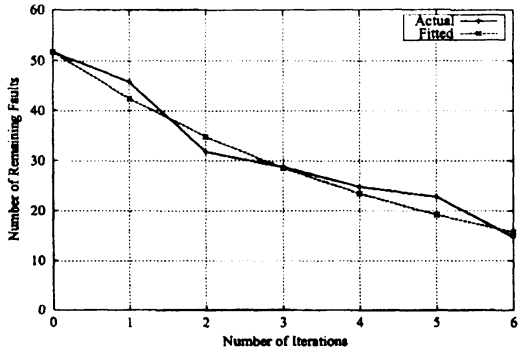

図 3： 推定された期待残存フォールト数 $a - \hat{H}_n$

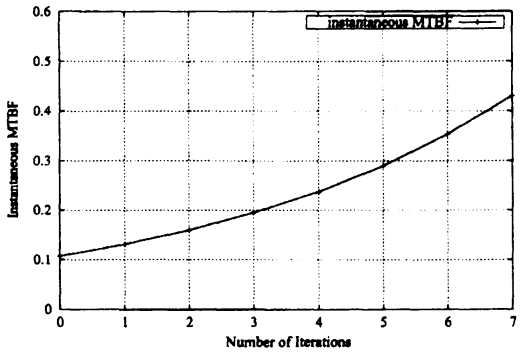

図 4： 推定された瞬間 MTBF

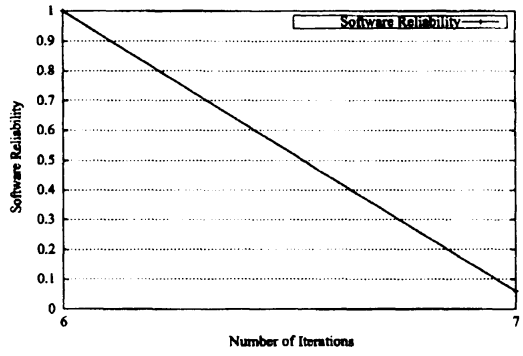

図 5： 推定されたソフトウェア信頼度

一方，それらの差が 1 未満である場合，比較対象の SRGM に優位性は見られず，推定が容易である自由パラメータ数の少ないモデルを最適 SRGM として扱う．MSE および AIC 基づく適合性評価の結果を表 4 および表 5 に示す．

MSE に基づく適合性評価を行った結果，DIS は全てのデータセットに対してパラメータの推定結果が得られ，高い適合性がみられた．また，離散化 NHPP モデルは連続型 NHPP モデルと比べて適合性の高い結果が安定してみられたため，アジャイル・ソフトウェア開発の信頼性評価において離散化 NHPP モデルの有用性が確認された．

MSE および AIC に基づいて連続型 NHPP モデルを比較した結果，全てのデータセットに対して DSS に最良の適合性がみられた．Project A および B においては，開発規模をテスト時間の代替メトリクスとした DSS が最良の適合性を示し，Project C においては，テストケース数をテスト時間の代替メトリクスとした DSS に最良の適合性がみられた．また，Project D の適合性評価の結果，DSS に最良の適合性がみられたが，MSE においてはイテレーション数が最良の適合性をみせ，AIC においてはテスト工数が最良の適合性をみせるという異なる結果が得られた．しかし，両者を比較した結果，MSE の値にほとんど差はなく，AIC においては優位性がみられなかったため，両者とも Project D の信頼性評価に有用なメトリクスであると考えることができる．

4.4 適用例

ここでは，Project A における信頼性評価の結果を一例として示す．離散化 NHPP モデルの中で最良の適合性をみせた DEXP の平均値関数 H_n の推定値および発見された総フォールト数の総計を図 2 に示す．図 3 に推定された期待残存フォールト数 $a - \hat{H}_n$ を示す．図 3 は Iteration 6 の開発を終えた時点で約 16 個

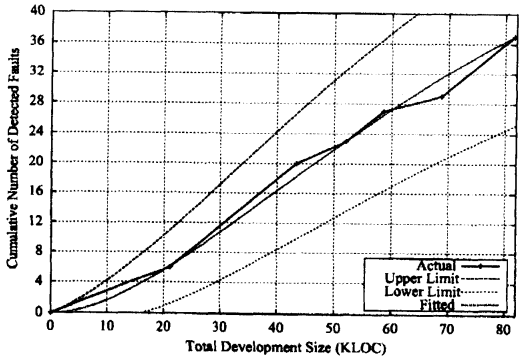

図 6： 推定された平均値関数 $\hat{H}(j)$

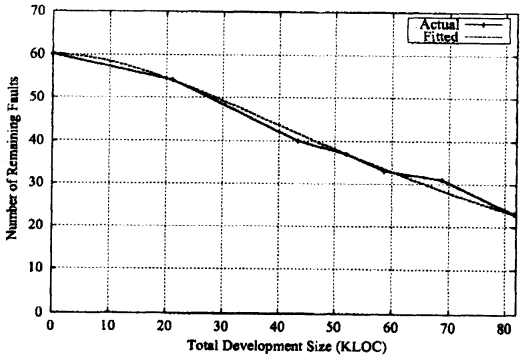

図 7： 推定された期待残存フォールト数 $a - \hat{H}(j)$

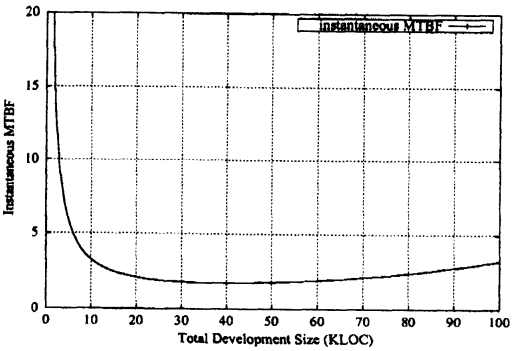

図 8： 推定された瞬間 MTBF

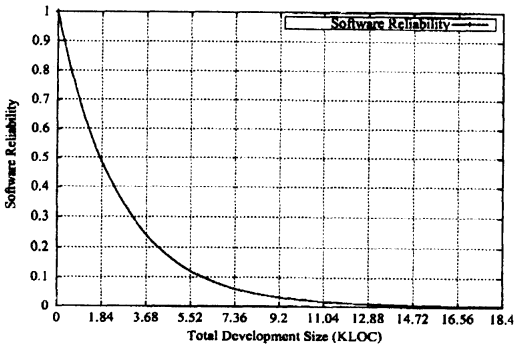

図 9： 推定されたソフトウェア信頼度

のフォールトが潜在していることを表している．さらに，Iteration 7 の開発が行われることを仮定し，図 4 および図 5 に推定された瞬間 MTBF およびソフトウェア信頼度を信頼性評価尺度として示す．

また，連続型 NHPP モデルの中で最良の適合性をみせたモデルとして，開発規模をテスト時間の代替メトリクスとした DSS の平均値関数 $H(j)$ の推定値，発見された総フォールト数の総計，および 90%信頼限界を図 6 に示す．図 7 に推定された期待残存フォールト数 $a - \hat{H}(j)$ を示す．図 7 より，Iteration 6 の開発を終えた時点で約 23 個のフォールトが潜在していることがわかる．さらに，Iteration 7 の開発が 18.4KLOC であると仮定し，図 8 および図 9 に推定された瞬間 MTBF およびソフトウェア信頼度を信頼性評価尺度として示す．

5 むすび

本論文では，アジャイル・ソフトウェア開発のプロセス計測データに重回帰分析を適用することで，ソフトウェア製品品質に影響を及ぼす要因を明らかにし，ソフトウェア品質の定量的評価を行った．重回帰分析を適用した結果，プロセス計測データを開発規模当りに規準化したことが分析精度の向上に大きく関わった．

アジャイル・ソフトウェア開発で採取可能なプロセス計測データを用いて，離散化 NHPP モデルおよび連続型 NHPP モデルを適用し，適合性比較を行った結果，離散化 NHPP モデルは連続型 NHPP モデルと比べて適合性の高い結果が安定してみられたため，アジャイル・ソフトウェア開発の信頼性評価における離散化 NHPP モデルの有用性が確認された．テスト時間の代替メトリクスとして様々なメトリクスを用いて連続型 NHPP モデルを適用した結果，全てのデータセットに対して遅延 S 字形 SRGM に最良の適合性が

みられた．さらに，遅延S字形SRGMにおいては全てのメトリクスに対してパラメータを推定することができた．したがって，連続型NHPPモデルにおいては遅延S字形NHPPモデルが信頼性評価に有効であると考察できる．

今回の適用結果より，本論文で使用した離散化NHPPモデルはパラメータ推定にかかる手間が少なくかつ正確な推定を行うことが可能なため，実用面からも有用性が期待できる．さらに，重回帰分析および連続型NHPPモデルを適用した結果から，アジャイル・ソフトウェア開発における定量的品質評価には，開発規模が重要なメトリクスとなると考えることができる．プログラムコードの量および複雑度と密接な関係をもつ開発規模がソフトウェア品質と深く関係したため，このような結果が得られたと推察できる．今後の課題として，ソフトウェア信頼性評価において，イテレーション数がさらに増加した状態のプロジェクトデータを用いて，推定値の適合性を確かめる必要性がある．

謝辞

本研究の一部は，日本学術振興会科学研究費補助金 基盤研究 (C)（課題番号 18510124）の援助を受けたことを付記する．

参考文献

[1] 好川哲人，鈴木道代：“PIMBOKか？アジャイルか？プロジェクト管理のための2つのアプローチ，システム開発ジャーナル，Vol. 3, pp. 19–43(2008).

[2] 永田靖，棟近雅彦：「多変量解析法入門」，サイエンス社，東京 (2001).

[3] 山田茂：「ソフトウェア信頼性モデル」，日科技連出版社，東京 (1994).

[4] 山田茂，福島利彦：「品質指向ソフトウェアマネジメント」，森北出版，東京 (2007).

[5] 藤原隆次，山田茂：“アジャイル開発環境におけるソフトウェア信頼性評価に関する一考察”，日本オペレーションズ・リサーチ学会春季研究発表会アブストラクト集，pp. 40–41(2007).

[6] 山田茂，井上真二，佐藤大輔：“ソフトウェア信頼性評価のための差分方程式に基づく統計的データ解析モデルに関する考察”，日本応用数理学会論文誌，Vol. 12, No. 2, pp. 77–90(2002).

[7] 山田茂，大寺浩志：「ソフトウェアの信頼性」，ソフト・リサーチ・センター，東京 (1990).

[8] 山田茂，藤原隆次：「ソフトウェアの信頼性：モデル，ツール，マネジメント」，プロジェクトマネジメント学会 (2004).