

개발과 유지보수 프로젝트의 이상적인 팀 규모

이 상 운[†] · 박 중 양^{††} · 박 재 흥^{†††}

요 약

팀의 규모에 관한 연구 결과 전문가들은 일반적으로 업무를 수행하는데 규모가 작은 팀이 큰 팀 보다 좋다는데 동의하고 있다. 또한, 항상 최소한의 실질적인 팀 규모를 구성하는데 목표로 하고 있다. 그러나 “작은”이 얼마를 의미하는지가 문제가 된다. 본 논문에서는 프로젝트 인도율과 인도속도와 같은 생산성 측도를 도입하여 이상적인 팀 규모를 결정하였다. 300개 프로젝트를 대상으로 데이터 집합을 2개의 서브 그룹인 개발 프로젝트와 유지보수 프로젝트로 분류하였다. 실험결과 가장 성공적인 프로젝트는 최대 5명으로 수행하는 경우였다. 본 논문은 실제 개발된 프로젝트들을 대상으로 생산성-기반 이론을 적용시켜 이상적인 팀 규모를 판단하는 지침을 제공한다.

An Ideal Team Size in Development and Maintenance Project

Sang-Un Lee[†] · Joong-Yang Park^{††} · Jae-Heong Park^{†††}

ABSTRACT

Experts work teams generally agree that small teams are better than large teams for getting things done. Also, always aim for the lowest practical team size. But what does “small” mean? We introduce to productive metrics such as project delivery rate (PDR) and speed of delivery (SOD) for decision criteria of ideal team size. The experimental is based on the analysis of 300 project data. These data sets are divide in two subgroups. One is a development project ; the other is a maintenance project. As a result of experimental in two subgroup, we come to the conclusion that the most successful projects has small teams indeed staffed with maximum five persons. This paper presents ideal team size in order to provide information that can be used as a guide in selecting the most practical and productive-based team size for a software development project.

키워드 : 기능점수(Function Point), 소프트웨어 규모(Software Size), 팀 규모(Team Size), 인도 속도(Speed of Delivery), 프로젝트 인도율(Project Delivery Rate)

1. 서 론

최근 들어, 소프트웨어 개발일정(기간)과 비용 추정 결과는 크게 부정확하며, 불충분한 품질을 갖고 있고, 개발 생산성은 사용자가 요구하는 수준보다 매우 느리게 향상되고 있다. Arthur[1]는 이러한 소프트웨어 개발 실상을 소프트웨어 위기(Software Crisis)라고 하였다. 이와 같은 이유로 인해, 프로젝트 관리 측면에서 소프트웨어 개발 및 유지보수 비용을 줄이고자 체계적인 연구가 수행되고 있으며, 소프트웨어 비용 산정, 소요 인력 및 개발일정을 추정하는 모델을 개발하는 계기가 되었다. 계획단계에서 보다 정확한 추정은 프로젝트를 관리할 때 발생하는 다양한 의사결정, 소요 예산과 개발인원 할당과 계약체결 여부에 신뢰할 만한 정보를 제공한다.

적기에, 주어진 예산범위 내에서 좋은 품질을 가진 소프트웨어를 개발하기 위해, 소프트웨어에 대한 가장 실제적이고 생산적인 팀의 규모를 선정하는 것이 관리자에게는 필수적으로 요구되는 지식이며, 가장 어려운 문제이다.

소프트웨어 이외의 타 분야에서 프로젝트 수행을 위한 팀 규모에 관한 연구로는 Orsburn et al.[2], Peters[3], O'Donnell [4], Katzenback et al.[5], Varney[6], Byram[7] 등이 있다. 또한, 소프트웨어 개발분야에서의 프로젝트 수행 팀 규모에 관한 연구로 정보시스템 개발에 관한 팀 규모 연구는 Samprevivo [8], LOC(Line Of Code)를 대상으로 연구한 사례는 Putnam et al.[9]이 있으며, 기능점수(FP, Function Point)를 대상으로 한 연구사례는 ISBSG(International Software Benchmarking Standards Group) Benchmark Release 6[10]가 있다.

규모가 작은 팀이 규모가 큰 팀보다 업무 수행 효율성 측면에서 보다 좋다는 것은 다양한 연구결과로 밝혀진 사실이다. 그러나 “작은”이 과연 얼마를 의미하는지가 문제로 대두된다. Putnam et al.[9]은 동일한 LOC를 대상으로 팀의

※ 본 연구는 2002년도 경상대학교 학술연구조성 지원비에 의하여 수행되었음.
[†] 정 회 원 : 국방품질관리소 항공전자장비 및 소프트웨어 품질보증 담당
^{††} 정 회 원 : 경상대학교 통계학과 교수
^{†††} 정 회 원 : 경상대학교 컴퓨터과학과 교수
 논문접수 : 2002년 6월 5일, 심사완료 : 2002년 9월 24일

규모를 결정한 사례를 연구하였으며, 이 연구는 LOC 측도에 대한 문제점을 내포하고 있다. 이에 반해 ISBSG Benchmark Release 6[10]는 소프트웨어 규모 측도로 기능점수를 사용하여 언어와 독립적이며, 일반적으로 적용할 수 있는 방법이다. 그러나 이 방법은 소프트웨어 규모별로 실제 수행된 팀의 규모 데이터의 평균 또는 중앙값만을 이용하여 팀의 규모를 제시하였다. 따라서, 보다 이론적인 근거에 기초한 팀의 규모를 제시할 필요성이 있다. 이에 따라 본 논문은 팀의 생산성 측도에 기반하여 이상적인 팀의 규모를 결정하고자 한다. 본 논문의 결과는 소프트웨어 개발시 소프트웨어에 대한 가장 실제적이고 생산적인 팀 규모를 선택하는데 지침으로 사용될 수 있는 정보를 제공한다.

2장에서는 소프트웨어 팀 규모 추정에 관한 관련 연구를, 3장에서는 이상적인 소프트웨어 팀 규모에 대한 ISBSG Benchmark Release 6[10]의 연구 결과를 고찰해본다. 4장에서는 생산성 측도 기준에 근거하여 이상적인 팀 규모를 제시하고, 제안된 팀 규모와 기존 연구 결과를 비교 분석한다.

2. 관련 연구 및 연구동기

타 분야에서의 프로젝트 수행을 위한 팀 규모에 관한 연구로는 Orsburn et al.[2], Peters[3], O'Donnell[4], Katzenback et al.[5], Varney[6], Byram[7] 등이 있다. Orsburn et al.[4]은 자율규제 팀은 평균적으로 6~18명의 고도로 훈련된 요원으로 구성되어 있을 경우 최대의 책임감으로 잘 정의된 업무를 수행할 수 있음을 보였다. Peters[3]는 팀은 아니지만 일본의 기본적인 조직 단위는 12명이며, 수 백년에 걸쳐 군대의 분대원은 8명이 기본원리임을 제시하였다. O'Donnell[4]은 시험 대상 샘플에 관해 토의하는 소비자그룹(Focus Group)은 8~12명으로 그룹을 구성하고 있음을 제시하였다. 또한, Katzenback et al.[5]은 30개 이상의 회사에서 50개 이상의 다른 팀에 있는 수 백명을 대상으로 연구한 결과 실질적으로 모든 효과적인 팀은 2~25명 범위를 가져야 함을 제시하였다. 예로, 북부 Piggybacking 팀은 7명, Knight-Ridder 신문사 팀은 14명이었으며, 대부분은 10명 이하로 팀을 구성하고 있었다. Varney[6]는 팀의 구조와 규모에 대해 연구한 결과 생산적인 팀을 구성하는 경우 가장 바람직한 팀 규모는 약 5명임을 제시하였다. Byram[7]은 '자율적인 업무 수행 팀의 마법'에서 의회 운영위원회의 직무는 실제로 10~15명의 팀인 경우 명백하게 정의된 책임이 개인에게 할당될 수 있다고 제시하였다. Arizona 대학의 한 연구[11]는 도서관 프로젝트를 수행하는데 필요한 지식, 숙련도, 능력과 품질을 가진 프로젝트 팀의 규모는 5~7명으로 추천하고 있다.

소프트웨어 개발과 관련된 프로젝트 팀 규모에 관한 연구는 Samprevivo[8], Putnam et al.[9]과 ISBSG Bench-

mark Release 6[10]가 있다. Samprevivo[8]는 정보시스템 개발에 대한 팀 규모를 결정함에 있어서 팀의 규모는 팀 구조가 가장 중요한 관점이며, 그룹의 성취도와 문제 해결 능력에 있어서 팀 요원 개개인의 만족도에 영향을 받음을 밝혔다. 팀 규모와 개별 요원의 불만족 관계에 대해 연구한 결과 팀의 규모가 증가함에 따라 개별 요원의 불만족이 증가하였다. 즉, 작은 그룹(5~7명)에서 개별요원은 가장 큰 만족을 느끼고 있으며, 큰 그룹(12~15명)에서는 가장 작은 만족을 느낌을 제시하였다. Putnam et al.[9]은 다음과 같은 기준을 제시하였다: (1) 작은 규모로 적절히 선택된 팀은 성공적인 프로젝트를 위해 필수적이다; (2) 실제로 큰 규모의 팀이 개발한 프로젝트보다 작은 팀이 적은 노력으로 보다 빨리 프로젝트를 종료한다; (3) 20명 또는 보다 많은 팀 요원으로 개발된 프로젝트는 5명 또는 그 이하의 요원으로 개발된 프로젝트보다 많은 노력이 필요하다. 이는 동일한 SLOC(Source LOC)를 갖는 프로젝트 비교로부터 유도되었다. ISBSG Benchmark Release 6[10]는 다음과 같은 기준을 제시하였다: (1) 작은 팀은 프로젝트 규모와 팀 규모간에 단순 관계를 갖고 있다; (2) 2~3명의 팀은 팀 요원 당 가장 큰 규모의 프로젝트를 개발하는데 적합하다; (3) 항상 가장 작은 실제 팀 규모를 목표로 삼는다; (4) 팀 규모가 5명을 초과하면 생산성은 감소하며, 작은 팀이 보다 생산적이다. 5 또는 그 이상의 팀 규모로 개발된 프로젝트는 보다 작은 팀으로 개발된 프로젝트보다 기능점수 1점을 개발하는데 특히 많은 노력이 소요된다. 또한, 프로젝트 설계시 "6명은 이상적인 프로젝트 팀 규모가 아니다"라는 주간 칼럼[12]에서는 가장 성공적인 프로젝트는 1명, 2명 또는 3명의 적은 팀으로 이루어지며, 완전한 팀 규모는 2~3명이라고 제시하였다. 또한, 4~5명도 괜찮은 규모이지만 6명은 혼잡하다고 제시하고 있다.

이와 같이 팀 선택 및 할당에 대한 연구 결과를 토대로 할 때, 성공을 위한 절대적 필요성보다는 실용적인 지침으로 팀의 규모를 보다 작게 하는 것은 명백한 사실이다. 또한, 실제로 5명 이하의 규모로 팀을 구성하면 팀 요원 간 원활한 의사소통과 최적의 생산성을 보이며, 적기에 소요 비용 범위 내에서 성공적으로 프로젝트 개발을 종료할 수 있음을 알 수 있다. 소프트웨어를 개발할 경우에도 이와 같은 원칙에 일치하는지를 평가할 필요가 있다.

3. ISBSG Benchmark Release 6[10]의 이상적인 소프트웨어 팀 규모 연구 결과

ISBSG Benchmark Release 6[10]는 기능점수와 팀 규모(TS, Team Size)를 알고 있는 244개 프로젝트를 대상으로 분석한 결과 <표 1>과 <표 2>의 결과를 얻었다. 여기서 팀의 규모는 프로젝트 수행 전 과정에서 프로젝트에 가장

많은 인원이 참여한 시점을 기준으로 파악된 인원이다. 따라서, 분석, 설계, 프로그래머, 시험, 품질보증 또는 관리자 등 어느 시점에서 어떤 요원들이 얼마만큼 참여하였는지는 세부적으로 알 수가 없다.

<표 1> 프로젝트 규모 FP에 따른 팀 규모 TS

기능점수 (FP)	표본 수	팀 규모 (TS)			
		Min	Max	Median	Mean
1~100	34	1	11	3	3.9
101~200	41	1	53	4	5.8
201~300	46	1	14	4	5.0
301~400	29	1	21	5	6.2
401~500	12	2	8	5	5.1
501~600	10	4	17	7	8.4
601~700	4	3	20	4	7.5
701~800	7	3	12	4	5.0
801~900	9	2	7	3	4.4
901~1,000	4	4	17	8	9.0
1,001~1,100	4	3	6	4	4.3
1,101~1,200	4	2	12	4	5.5
1,201~1,400	9	2	16	4	6.9
1,401~1,600	7	3	65	7	15.4
1,601~1,800	4	5	8	7	6.8
1,801~2,000	3	3	25	11	13.0
2,001~3,000	8	3	26	11	12.5
3,001~4,000	3	8	15	10	11.0
4,001~5,000	3	12	50	24	28.7
5,001 이상	3	8	10	8	8.7

<표 2> 팀 규모에 따른 프로젝트 규모

팀 규모 (TS)	표본 수	기능점수 (FP)			
		Min	Max	Median	Mean
1	19	25	350	82	115
2	33	11	1336	236	327
3	35	39	2145	297	565
4	30	39	1437	329	521
5	24	32	2014	291	499
6	19	33	1093	200	309
7	15	54	1694	499	679
8	15	204	9803	445	1930
9	7	154	1203	303	424
10	12	89	5789	234	1089
11	11	60	1882	374	620
12	5	231	4562	769	1443
13 이상	19	101	4913	1457	1621

실험 결과, 평균적인 팀 규모는 일반적으로 프로젝트 규모가 커짐에 따라 증가하였다. 그러나 Mean과 Median 값은 100 FP~1400 FP에서 4~6명이었으며, 그 이상의 기능점수에서는 실질적으로 팀 규모가 증가하였다. 또한, 작은 팀 규모에 대해서는 프로젝트 규모와 팀 규모 사이에 단순한 관계를 가지고 있으며, 2~3명의 팀 요원으로 구성했을 경우 생산성이 가장 크게 나타났다. 기능점수가 3,000 이상인 프로젝트는 프로젝트 규모를 표현할 수 있는 대표적인 표본이 나타나지 않는다. 경험적으로 볼때, 팀 규모가 30~50이면 기능점수 1점을 개발하는데 20~40 시간이 소요되었다.

이는 팀 규모가 증가하면 생산성은 감소하며 인도기간도 지연됨을 의미한다. 기능점수가 2,000점 이상인 프로젝트에 대해 팀 규모 관리 요소와 프로젝트의 위험도가 증가함에 따라 <표 3>의 일반화된 결과를 제시하였다.

ISBSG Benchmark Release 6[10]에서 얻은 위의 결과는 단지, 기능점수 규모에 대해 투입된 팀의 평균값에 대한 기준으로 제시된 것이다. 따라서 보다 이론적인 근거를 바탕으로 이상적인 팀의 규모를 결정하는 기준이 필요하다. 이 기준을 만족하기 위해 본 논문은 생산성에 기초하여 이상적인 팀의 규모를 결정하고자 한다.

<표 3> 일반적인 팀 규모 선정 결과

기능점수 (FP)	팀 규모 (TS)		
	Min	Max	Mean
1~300	1	6	3~6
300~500	2	10	5
500~1,000	4	12	6~8
1,000~1,500	5	15	5~7
1,500 이상	7	25	10

4. 이상적인 팀 규모 결정

4.1 평가 기준

이상적인 팀의 규모를 결정하기 위해 프로젝트 규모에 대한 팀 규모의 평균값 또는 중앙값을 취하는 ISBSG Benchmark Release 6[10]의 방법 대신 생산성 측도(Metrics) 기준을 적용함으로써 보다 이론적인 근거를 바탕으로 최적의 팀 규모에 대한 지침을 제공하고자 한다.

평가측도 : 이상적인 팀의 규모를 결정하기 위해 생산성 측도인 PDR과 SOD 평가 측도를 이용한다.

ISBSG Benchmark Release 6[10]는 소프트웨어 기능을 사용자에게 양도하는 비율인 프로젝트 인도율(PDR, Project Delivery Rate)을 개발노력의 계수로 측정하였다. 즉, PDR은 소프트웨어의 규모인 기능점수 FP와 개발에 투입되는 노력(E, Effort)로 정의하였으며, $\frac{E}{FP}$ 로 계산하였다.

이는 기능점수 1점을 개발하는데 소요되는 노력이다. 또한, 인도속도(SOD, Speed of Delivery)는 주어진 기간(D, Duration) 동안 팀이 소프트웨어 기능점수 FP를 인도하는 능력으로, $\frac{FP}{D}$ 로 계산하였다.

생산성 측면에서 볼 때, 이상적인 팀의 규모는 1명당 FP 1점을 개발하는데 소요되는 노력이 최소가 되고, 매월 인도되는 FP가 최대가 되는 기준을 적용하여 선정하는 것이 바람직하다.

평가기준 : 팀 요원 1명당 PDR이 최소가 되고 SOD가 최대로 되는 팀을 이상적인 팀으로 결정한다.

개발 프로젝트에 대해 실제로 수행된 팀 범위 중에서 최소의 팀에 대한 생산성이 1 순위 (***)는 11개, 2 순위 (***)는 3개, 3순위 (*)는 5개를, 기타 1개(1001~1100 FP)를 기록하였다. 유지보수 프로젝트의 경우 501~700, 801~1000, 1001~5000 이상 범위는 1개의 팀만 있는 경우와 전혀 수행된 실적이 없는 경우가 있어 이상적인 팀 규모를 판단할 수가 없다. 따라서, 나머지 7개의 FP 범위에 대해서만 고려해 보자. 유지보수 프로젝트에 대해서는 팀 범위 중에서 최소의 팀에 대한 생산성이 1 순위는 7개중에서 4개, 2 순위는 2개, 기타 1개(1~100 FP)를 기록하였다. 따라서, 실제적으로 수행된 프로젝트 팀 규모의 데이터를 종합해 볼 때, 다음과 같은 결론을 얻을 수 있다.

결론 1 : 반적으로 최소의 팀 규모로 수행된 프로젝트들이 가장 좋은 생산성을 갖고 있음을 알 수 있다.

생산성 측면에서 상위 3개의 순위를 가지는 팀의 규모를 토대로 볼때, 개발과 유지보수 프로젝트에 대해 프로젝트 규모별로 이상적인 팀 규모를 <표 11>에 제시하였다. 일반적으로, 개발 프로젝트는 $1 \leq FP \leq 400$ 은 1명, $401 \leq FP \leq 900$ 은

2명, $901 \leq FP \leq 3000$ 은 3명이 적합함을 알 수 있으며, $3001 \leq FP$ 는 8명 이하의 팀으로 개발된 프로젝트가 없어 실제 데이터를 근거로 할 경우 최적 인원은 8명이 적당하나 5명으로 이상적인 팀을 구성하여도 무방할 것으로 보인다. 이에 반해, 유지보수 프로젝트는 대부분이 작은 규모의 프로젝트를 수행하였으며, $1 \leq FP \leq 400$ 은 1~2명, $401 \leq FP \leq 800$ 은 3명, $801 \leq FP \leq 1100$ 은 4명, $1001 \leq FP$ 는 5명이 적합함을 알 수 있다. ISBSG Benchmark Release 6[12]는 단순히 기능 점수에 따른 평균적인 팀의 규모를 6명에서 10명까지 제시하였다. 이에 비해 본논문은 실제 개발된 프로젝트들을 대상으로 생산성-기반 이론을 접목시켜 이상적인 팀 규모를 판단하는 기준을 제안하였다. 또한, 제안된 팀 규모는 소프트웨어 분야의 팀 규모 연구 결과[8-10]와 일치하였다. 따라서, 추가적으로 다음의 결론을 얻을 수 있다.

결론 2 : 소프트웨어 팀을 구성함에 있어 가장 성공적인 프로젝트는 1, 2, 또는 3명의 작은 팀으로 구성되어 있으며, 5명은 소프트웨어 개발에 있어서 절대적인 최대 인원이다.

<표 11> 프로젝트 규모별 이상적인 팀 규모 비교

FP	이상적인 팀 규모		
	제안된 팀 규모		ISBSG Benchmark Release 6 제시
	개발 프로젝트	유지보수 프로젝트	
1~100	1	1~2	1~6
101~200	1	1~2	1~6
201~300	1	1~2	1~6
301~400	1	1~2	5
401~500	2	3	5
501~600	2	3	6~8
601~700	2	3	6~8
701~800	2	3	6~8
801~900	2	4	6~8
901~1000	3	4	6~8
1001~1100	3	4	5~7
1101~1200	3	5	5~7
1201~1400	3	5	5~7
1401~1600	3	5	10
1601~1800	3	5	10
1801~2000	3	5	10
2001~3000	3	5	10
3001~4000	5	5	10
4001~5000	5	5	10
5000 이상	5	5	10

5. 결 론

소프트웨어 개발시 프로젝트를 원활히 개발하기 위해 팀 규모를 결정하는 것은 프로젝트 관리자에게 필수적으로 요구되는 정보이다. 타 분야나 소프트웨어 개발분야에 대해 일반적으로 이상적인 팀 규모는 5~7명 또는 5명 이하가 가장 적합함을 알 수 있었다. ISBSG Benchmark Release 6 [10]는 기능점수 규모별 팀 규모에 대한 평균 또는 중앙값을 이용하여 이상적인 팀 규모를 제시하였으나 기능점수 규모가 작거나 큰 모든 범위에서 이상적인 팀의 규모를 5~10명까지 제시하였다.

본 논문의 실험에는 ISBSG Benchmark Release 6[10]의 300개 프로젝트가 사용되었으며, 이들 데이터 집합을 개발 단계와 유지보수단계에 적용된 프로젝트들로 분류하여 팀의 생산성 측도인 PDR과 SOD를 고려하여 1명당 PDR이 최소가 되고 SOD가 최대가 되는 기준을 적용하여 이상적인 팀 규모를 선정하였다. 실험 결과 개발단계나 유지보수단계 프로젝트를 모두 소프트웨어의 규모에 따라 대부분이 3명 이하의 팀 구성이 요구되었으며, 최대 5명이 필요하였다. 이는 기존의 연구결과와도 일치하는 결과이다.

개발기간, 팀 구성원의 능력에 따라 소프트웨어 규모별 이상적인 팀 규모에 영향을 받을 수 있으나 본 논문은 단순히 소프트웨어 규모에 따른 이상적인 팀 규모만을 고려하였다. 따라서, 추후 이 분야에 대한 연구를 수행할 것이다.

참 고 문 헌

- [1] L. J. Arthur, "Measuring Programmer Productivity and Software Quality," New York, Jone Wiley, 1985.
- [2] J. D. Orsburn et al., "Self-Directed Work Teams : The New American Challenge," Business One Irwin, p.8, 1990.
- [3] T. Peters, "Thriving on Chaos," Knopf, p.297, 1987.
- [4] J. M. O'Donnell, "Focus Groups : A Habit-Forming Evaluation Technique," Training and Development Journal, p. 71, 1988.
- [5] J. Katzenback and D. Smith, "Discipline of Teams," Harvard Business Review, p.114, 1993.
- [6] G. Varney, "Building Productive Teams," Jossey-Bass, p. 115, 1990.
- [7] W. Byram, "Self-Directed Wo가 Team Magic," Boardroom Reports, p.8, 1991.
- [8] P. C. Semprevivo, "Teams in Information Systems Development," Yordon, p.85, 1980.
- [9] L. Putnam and W. Myers, "Selecting the Right Team Size : Small is Beautiful," Cutter Consortium, <http://www.cutter.com/consortium/research/1998/crb981222.html>, 1998.
- [10] ISBSG, "Worldwide Software Development - The Benchmark Release 6," Victoria, Australia International Software Benchmarking Standards Group, 2000.
- [11] "Project Team Appointment Process," <http://dizzy.library.arizona.edu/library/teams/slrp/syllabus/pt-appoint.html>.
- [12] "Six, not the Ideal Project Team Size," A Weekly Column on Software Design, <http://www.sum-it.nl/en199947.html>, 1999.



이 상 운

e-mail : sangun_lee@hanmail.net

1983년~1987년 한국항공대학교 항공전자공학과(학사)

1995년~1997년 경상대학교 컴퓨터학과(석사)

1998년~2001년 경상대학교 컴퓨터학과(박사)

1992년~현재 국방품질관리소 항공전자장비 및 소프트웨어 품질보증 담당

관심분야 : 소프트웨어 공학(소프트웨어 시험 및 품질보증, 소프트웨어 신뢰성), 소프트웨어 메트릭스, 신경망, 뉴로-퍼지



박 중 앙

e-mail : parkjy@nongae.gsnu.ac.kr

1982년 연세대학교 상경대학 응용통계학과(학사)

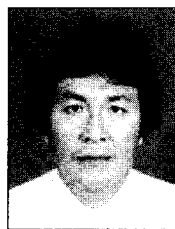
1984년 한국과학기술원 산업공학과 응용통계전공(석사)

1994년 한국과학기술원 산업공학과 응용통계전공(박사)

1984년~1989년 경상대학교 전산통계학과 교수

1989년~현재 경상대학교 통계정보학과 교수

관심분야 : 소프트웨어 신뢰성, 신경망, 선형통계 모형, 실험 계획법 등



박 재 흥

e-mail : pjh@nongae.gsnu.ac.kr

1973년~1978년 충북대학교 수학과(학사)

1978년~1980년 중앙대학교 대학원 전산학과(석사)

1985년~1988년 중앙대학교 대학원 전산학과(박사)

1983년~현재 경상대학교 컴퓨터학과 교수

관심분야 : 소프트웨어 신뢰성, 시험도구 자동화, 시스템 분석 및 설계, 신경망