Sci.Int.(Lahore),25(4),131-137, 2013 ISSN 1013-5316;CODEN: SINTE 8

THE ADAPTIVE AGILE IN HIGH PERFORMANCE COMPUTING: CASE STUDY ON PARALLEL IMAGE FEATURE EXTRACTION

Deshinta Arrova Dewi
(Universiti Kebangssan Malaysia, Bangi, Malaysia
deshinta_a@yahoo.com)
Elankovan Sundararajan
(Universiti Kebangssan Malaysia, Bangi, Malaysia
elan@fskm.ukm.my
Anton Satria Prabuwono
(Universiti Kebangsaan Malaysia, Bangi, Malaysia
antonsatria@fskm.ukm.my)

ABSTRACT: *The adaptive agile had been proposed by many practitioners to satisfy the unpredictable requirements that normally happen in the development of computer systems. Since then, it became a feasible choice for large scale research development (R & D) project including HPC. HPC products are essentially R & D product means it constantly drives system capability boundaries to upmost level in term of speed, performance or functionality. The R&D projects afforded the time to be managed with iteration model that designed for repeated cycles of defining objectives, alternatives, execution, analyzing results and risks. The adaptive agile for high performance computing involves its ability to rapidly model, handle processes to see the influence of numerous decisions before it is made. In this view, the adaptive agile has offered consideration for both efficiency and flexibility in adopting repetition and changes in any stage of development.*
The purpose of this paper is to study how the adaptive agile can support HPC development through best practices of popular agile i.e. SCRUM methods and eXtreme Programming (XP). We selected a case study on parallel image feature extraction since this system is intensively dynamic from distributing image(s) into segments, detecting image overlaps, controlling distributed segments and performing redistributed segments to extract the features of the image(s). This paper shows the HPC infrastructure that meets the nature of agile computing that bring many benefits compared with traditional methodology.

Key words: Adaptive Agile, High Performance Computing, Parallel Image Feature Extraction

1. INTRODUCTION

The agile method focuses on the flexibility to adapt with variety of changes with less documentation. Unlike the traditional method that documentation-driven throughout the life cycle, the agile encourages on the processes of finding what is right for the software development and document them accordingly [1,7,8,9]. By having this kind of development, the agile team can concentrate towards the possible solutions to solve problems, quickly experiment them and record the processes. This characteristic makes the agile method popular in industry environments since it truly welcomes changes in all stages throughout a software life cycle. [2,3,4,5,6]

In High Performance Computing the massive changes towards computing requirements during development stage happen in order to achieve maximum performance level at minimum cost. Dell Company even combines multiple systems to execute complex calculation for the sake of meeting customers' demand. The current HPC technology concentration is on emerging parallel processing algorithms and systems by integrating both administration and parallel computational techniques. The HPC then transports the collection of numerous technologies for instance computer architecture, algorithms, programs and electronics, and system software to resolve unconventional problems effectively and rapidly. An efficient HPC system involves a high-bandwidth, low-latency network to attach several nodes and clusters. In general, faster processing is the most demanding application in HPC and they tend to be used for computational analysis, data-intensive research, rich media, three-dimensional computer modeling, seismic processing, data mining and large scale simulation. [23,24,25,26]

Driven by CPU-intensive processing, such applications handle large volumes of data over short periods of time while also in some cases approving simultaneous access from multiple servers. The need to process large volumes of data quickly has huge effects for HPC storage requirements; given that storage I/O capabilities are typically much lower than those of processors. An HPC storage system for example needs large capacity accessible at high speed and to be highly expandable, while offering a single global namespace accessible to all users involved in the project.

The HPC technology currently is implemented in multidisciplinary areas such as biosciences, biographical data, oil and gas industry modeling, electronic design automation, climate modeling, media and entertainment, and also image processing systems.

1.1 Image Processing Systems

The image processing systems have considered their application in the form of HPC by performing processes in parallel execution. Since then, the parallel computing is highly demand as competent solution to improve the performance. Nowadays, the requirements to do intensive computation processes towards some images that have been captured previously by the other system are rising. For instance, in cold rolling strip surface defect inspection system that relies on its computer vision to capture images has caught the quantity of huge image processing that gradually created a bottleneck. It limits the performance of the overall system therefore the parallel system has come to an option to solve this issue. With specific comparison on the strategy and architecture to decide which one that actually can achieve the better efficiency. [17,18,19]

The HPC always deal with the architecture choices. The most common architectures of parallelization that normally suggested by the researchers are multicore CPUs and many-core graphics processors (GPU). Both offer two distinct features. Multicore CPUs offer great flexibility and are able to accommodate variety of parallelization strategies. Many-core GPU offers a huge number of cores but their peak efficiency is limited to data parallel applications. To perform the comparison there are many options to explore for instance a canny edge detector involved as benchmark of the experiments. The result state the strategy that well runs on a specific environment it may be performing poorly in another. The agile approach definitely helps this process by adopting its iterative and incremental development that promotes foreseen interactions in finding the best solution. [23,25]

Since applications have increasing complexity and present a wide-ranging diversity of parallel features, the complexity is even higher to decide which parallelization strategy is proper for a specified architecture to reach highest performance. The agile methodology by right is the collection of methods that understands how dynamic the development of processes is. The agile believes that the only constant in the software methodology is changes. Therefore, some researchers are strongly suggested the agile methodology to develop an HPC system.

This paper proposes a security-enhanced design maintenance process through an approach that makes use of misuse cases related to integrity, and similar variants, to analyze secure software architecture.

2. Contribution Summary

This paper presents the fact that the repetitions of processes do help the development of parallel system. While the agile actually works more towards people oriented, the awareness of the dynamic requirements and changes help the team to adopt new situation faster.

As addition to that, the detailed work is presented on how the adaptive agile promotes the dynamic documentation style for the ease of the developers to fulfill their deadline. While previous works published on agile methodology on the early stage of HPC system development, this paper explains most stages where the adaptive agile can give effective contribution to it. The purpose is such to give an overview that parallel image feature extraction is suitable with the adaptive agile hence it is brought here in this research.

This paper recommends two major contributions:

i. We introduce the agile approaches to a case study in HPC i.e. Parallel feature extraction.

ii. We introduce variety of the agile model that provide solution to the issues that usually met by the HPC system.

3. Problem Statement

The detailed issues that usually met by the HPC can be summarized as follows:

1. Accommodating constant changes in both problem specification and target architectures as computational methods and architectures evolve.
2. The large gap between HPC design and implementation models in application development.
3. Achieving high performance for a single application on different HPC platforms.

The key of the detailed issues is to provide the best adaptive development model that enables the system to be completed in shorter time and lower cost. All of those problems can be addressed properly through refinements. Since this paper specifically uses parallel feature extraction as a case study, the above main issue will be deepened with detailed problems that normally found in the parallel feature extraction for example the effects of algorithms migration from sequential to parallel environment, the architecture decision that make higher performance is possible, and so on. The adaptive agile methodology is expected to provide solution that covers the above concerns.

4. Related Works

Senad Cimic[27] in his paper stated that the agile methodology has helped challenges during requirement stages. He mentioned whenever requirements are not 100% defined most likely at some point of the lifecycle, some requirements may get changed. Or else, the new one could be familiarized. He then suggested Parallel Process Development Lifecycle (PPDL) to satisfy this situation. PPDL is actually has adopted agile methodology that is more adaptive than predictive[17,18,19]

There were past research works that have suggested some adaptive methodologies to support the development of HPC systems. Peter H Millis and team for instance, have come out with Proteus system, an application development methodology that elaborated the independencies of high level architecture through refinement processes. The processes involved were translation of the architecture detailed to programming notations via low level architecture specifications. To accommodate changes in specification, Proteus used a tree-structured development process that allows multiple architectures to be targeted by the implementation strategies. On top of that, Proteus also provides a systematic means for both changes in specification and target architecture. [21,23]

The researchers from Old Dominion University, Georgia State University and Nava Postgraduate School [19] have created a dynamic simulation model for agile software development. They stated that empirical research to evaluate the effectiveness and appropriateness of agile approaches is infrequent. The dynamic model deliberates composite interdependencies among the diversity of practices used in agile development. The validation was conducted through refactoring on project performance and the economic aspects of pair programming. They concluded that the dynamic model of agile approaches can be used as a tool to study agile development, and helps practitioners make better decisions and formulate appropriate software processes. In relation with the HPC, the development of HPC software can refer to this model also. [19]

On the other hand, the area of image processing system has brought the researchers to look into the development of applications that utilized the architecture of current processors i.e. multicore and GPU. It means the system has been upgraded to distributed environment. As consequences, all traditional or sequential algorithms have to be migrated to parallel algorithms that more suitable to run on distributed environment. [18,21]

As part of image processing systems, feature extraction is the key of image analysis because the processes enable the identification of object on the picture such as color, location, shape, texture, etc. Many research works have upgraded the sequential feature extraction algorithms to parallel algorithms. The main function of the feature extraction maintain same as a tool to retrieve quantifiable

Sci.Int.(Lahore),25(4),131-137, 2013 ISSN 1013-5316;CODEN: SINTE 8

properties of images. Just the algorithms run faster due to the parallel execution on distributed environment.

There are two types of features that retrieved the by feature extraction algorithms i.e. general features and domain-specific features. General features are the ones that found on images such as color, texture and shape of object on the images. According to the abstraction level, they can further divided into pixel-level features, local features and global features. The color and location of the objects are normally done in pixel-level, hence sometimes called pixel-level feature.[19,20]

The local features are referring to the objects that calculated over the results of sub division of the image band such as image segmentation or edge detection. The global feature refers to objects that calculated over the entire image or just regular sub-area of an image. The domain specific feature is application dependent features such as human faces, fingerprints and conceptual features, the identification of object on the picture such as color, location, shape, texture, etc.

Many research works have upgraded the sequential feature extraction algorithms to parallel algorithms. The main function of the feature extraction maintain same as a tool to retrieve quantifiable properties of images. Just the algorithms run faster due to the parallel execution on distributed environment.

There are two types of features that retrieved the by feature extraction algorithms i.e. general features and domain-specific features. General features are the ones that found on images such as color, texture and shape of object on the images. According to the abstraction level, they can further divided into pixel-level features, local features and global features. The color and location of the objects are normally done in pixel-level, hence sometimes called pixel-level feature.[19,20]

The local features are referring to the objects that calculated over the results of sub division of the image band such as image segmentation or edge detection. The global feature refers to objects that calculated over the entire image or just regular sub-area of an image. The domain specific feature is application dependent features such as human faces, fingerprints and conceptual features.

5. The HPC Software Development and The Adaptive Agile

The Software Development in HPC is non-trivial processes and requires thorough understanding of the application and the architecture. The parallelization strategy and specification should be defined after application analysis stage as stated in figure 1 below.

The HPC applications are well known as such applications that have time consuming characteristics before they arrive to a suggested solution. The common characteristics are long-running, taking hours even days to solve problems. The applications have to be stopped or deferred their computation processes in order to change codes, redeploy the latest updates and queue the scheduled processes to run.[26]

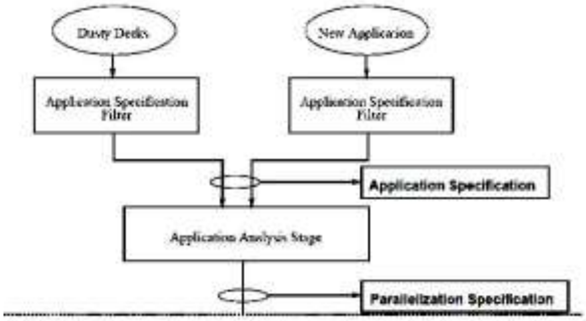


Figure 1 First Step of HPC Software Development

When these characteristics are not well handled, inefficiencies become big issues in industry environment such as wasting time of programmers and system analysts and also expensive computer resources. Many researchers suggest variety of approaches to save the programmers' time and utilize expensive computer resources productively. Hence, the second step in HPC Software Development is captured in figure-2 below that shows the Parallelized structure must be done right after the application development stage.

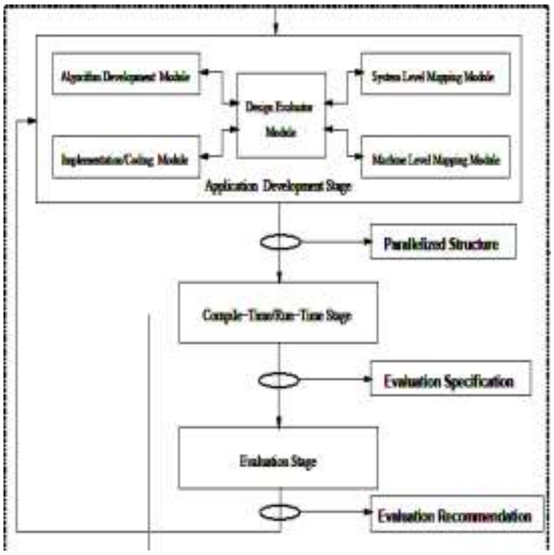


Figure 2 Second Step of HPC Software Development

The last step in HPC Software Development is Maintenance and Evolution stage that includes monitoring the operation of the HPC software and ensuring that it continues meeting its specification. The example of tasks in this step is correcting bugs as they surface, handle modifications needed to incorporate changes in system configuration, etc.

The last step in HPC Software Development is Maintenance and Evolution stage that includes monitoring the operation of the HPC software and ensuring that it continues meeting its specification. The example of tasks in this step is correcting bugs as they surface, handle

modifications needed to incorporate changes in system configuration, etc. According to the software assurance guide, functional specifications for security features are required by users. Examples include requiring user authentication to access specific data or requiring user assets before using privacy features.

6. The Agile Modeling for Parallel System Development

Agile Modeling (AM) is a methodology that relies on experiments and practices for the purpose of effective modeling and documentation of software systems. Hence, AM is mostly a series of practices that directed by principles and values for daily basis activities performed by software professionals. The AM does not express detailed procedures for how to produce a given type of model instead it offers recommendations for how to be effective as modeler. On the other hand, the eXtreme Programming (XP) is also an agile practice based methodology however it covers full development lifecycle. The XP therefore is base process into which the AM techniques may be tailored. [1,2,3,4]

The previous works showed that parallel programming that employs distributed memory of multiprocessors is the most challenging computing application fields due to its nature to coordinate series of distributed computing resources in order to solve problems and run them in parallel environment. Normally those problems are highly computational intensive. Distributed multiprocessors currently treated as the best problem solver in many domains of HPC, for example scientific simulation, bio informatics and image processing.

There are few models for applying agility to the parallel software development life cycle. One of the models is exposed in figure 3 below. This model enables the software developer to continue development while the software is in use. This model is also take consideration of four essential processes in parallel system development i.e. requirements, development, quality assurance and management. Those four processes are linked to each other and performed instantaneously.

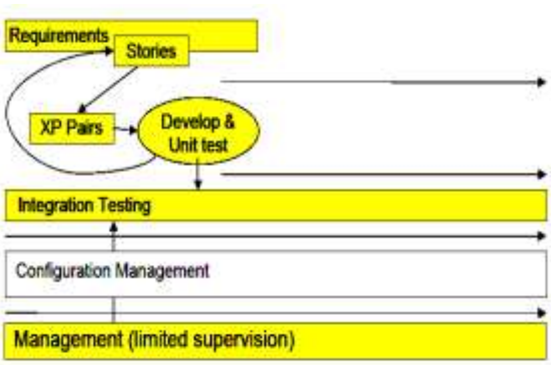


Figure 3 Adaptive Agile in Parallel Computing by Senad Cimic [27]

The XP approach is the most applicable agile methodology to parallel software development. As per seen in the figure 3, XP pairs become essential part of the iteration processes to determine the requirements and the development of the system with its unit testing. The other processes such integration testing and configuration management remain sequential. By having this kind of model, the documentation is reduced and definitely accelerates the development stage with lower cost. From the perspective of the team, the XP pairs that emphasize on the pair programming will benefit the team by having the knowledge spread among team members. This situation directly increases the speed of the whole processes. [6,8,9]

However this model has a weakness on the reduced documentation. It may lead to extreme difficulties while developing new software based on the earlier prototype with less information about it and more towards working features.

7. The Implementation of Parallel Feature Extractor

The parallel feature extraction in this research is actually part of the ongoing research in Centre of Artificial Intelligence Technology (CAIT) in National University of Malaysia (UKM) regarding distributed real time image processing systems for global vision. The experiments of using Scrum-XP model are promoted since the team met challenges in selecting appropriate methods to meet the system demand.

Currently the research reaches at the second stage of system analysis development and focusing on parallel image processing that able to extract information from series of pictures that were captured from the global vision system. The following table describes the case study that has been carried out towards the respective project. This case study is selected because it is the most important component that supports the overall system functions.

Table 1 Description of Case Study

Product Type	Parallel Feature Extraction
Size	Medium
Project Type	Average
Type of case study	Feature Extraction
Project duration	6 weeks
Programming Approach	Object Oriented
Feedback	Daily and weekly
Language	Java
Development environment	Java Netbean 7.3.1
Documents	Ms. Office 2010
Other tools	OpenCV and Rational Rose
Testing	Module and Unit
Reports	Main report
Web Server	N/A

The parallel modeling used in this research is based on the Nevatia-Babu contour-pixel detection algorithm and the linear approximation algorithm[18]. There are 3 major process steps involved: 1. Edge Detection, 2. Thinning and Thresholding, 3. Edge Linking. The input sent to contour-pixel detection is an image array of pixel that is generated from a 2D image. The expected output is a same sized array with directed contour-pixels that are hidden in the array. Those 3 major process steps are usually called as window operations.[18,20,24]

The figure 4 below shows parallel algorithms for Contour-pixel detection usually by partitioning *(n x n)* image array into *P* blocks of size (n/ $\sqrt{p}$ x n/ $\sqrt{p}$). The window operations will correspond to partitioned block in a node. The window operation can be done independently. The picture below depicts the detailed process in window operations. The independent window operation makes data parallelism can be exploited naturally.

Sci.Int.(Lahore),25(4),131-137, 2013 ISSN 1013-5316;CODEN: SINTE 8

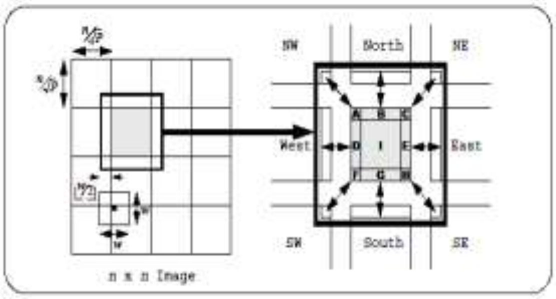


Figure 4 Data exchange between nodes for a boundary padding operation [18]

The linear approximation is actually the output of the contour-pixel detection. The data will be stored in contour pixel array that consist of 2D images.

There are two major steps in linear approximation i.e. local and global. The local contour is a contour whose pixels are placed in one single node while global contour pixels are placed in more than single node. The local contour is easier to be parallelized since it has no data dependencies between the nodes hence it can be done independently. The global contour requires neighboring node that completes the linear approximation. The processes then only can be done independently after neighboring nodes. [16,18,19,28]

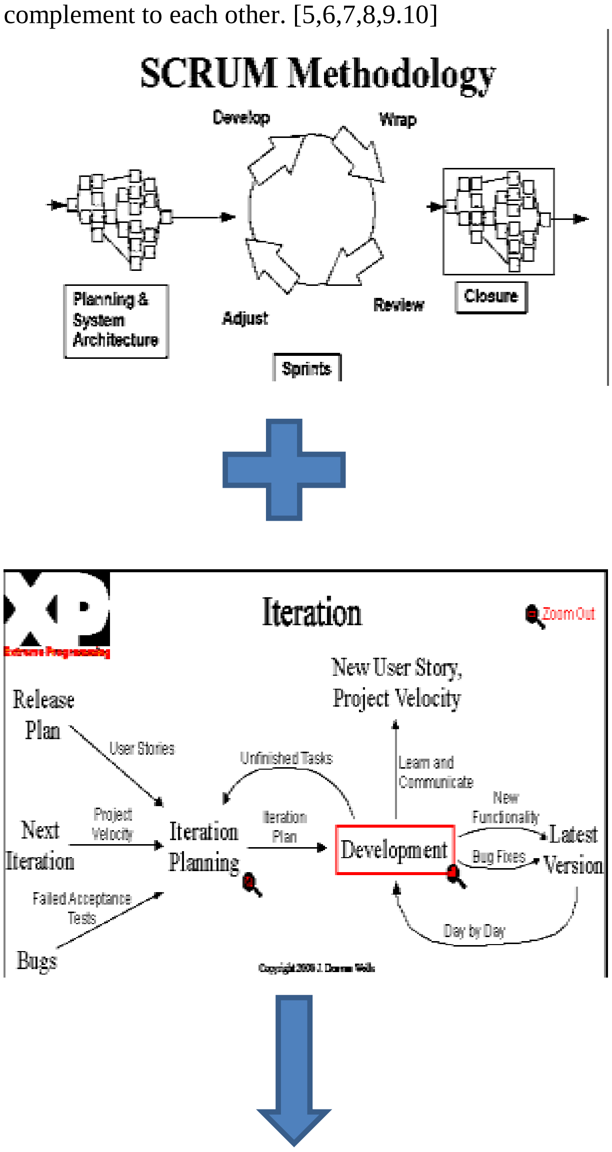
The Agile computing generally gives emphasis to adaptability and customization towards various system requirements. In HPC, the system requirements cover well understanding of the system application and the architecture. For example, during the course of software development, the developer is required to select the optimal hardware configuration for a particular application as well as the best decomposition and mapping of the problem onto selected hardware configuration. Apart from that, the developer is also to determine the best communication and synchronization strategy to avoid conventional problems in parallel systems such as bottle neck, deadlock and live-lock due to limited resources that are accessible by many processes.

To encounter the large gap between HPC design and its implementation, there are models which can be established to summarize the system design with detailed explanation of how the implementation takes place. In Table-2 the system requirement for region growing is tabulated according to functional and non-functional requirement that concern with technical difficulties that might be raised during system implementation. For example some region growing algorithms depend on the iteration stage and neighbor pixels. The accuracy produced by processes of neighbor pixels determines the effectiveness of the algorithms, hence, to parallelize them usually by using non straightforward approaches.

Table 2 Fragment of System Requirements

General Objectives: Partitioning images into 5 sub images horizontally and/or vertically
Functional Requirements: • **Concern: Simple division into Pi=5**
The system should be able to detect whether the partitioning may go across the sub-image borders that give difficulties of the algorithm to find direction to conduct region growing. The possible solution for this matter is to allow communication between sub-images that share overlapping at each iteration.
Functional Requirements: • **Concern: Depict overlapping needed to process each sub images**
The system should detect if communication between sub-images create the excessive numbers of communication. The possible solution for this matter is to conduct region growing for both vertical and horizontal and the resulting image is then obtained by combining both sets of sub-images.
Non Functional Requirements: • **Concern: Performance**
The system should accomplish the partitioning of region growing operation at a speed of 5,000 milliseconds for each data processes.

To accommodate constant changes in both specification and target architecture in HPC, the previous works had proposed solution that would be the adaptive agile model that combines XP and Scrum approach.[11,12,13] This combined model based on the strength and weakness of both XP and Scrum for example XP focuses on system implementation or codes while Scrum focuses on design.[14,15]Scrum lacks in team activities to complete iteration required by the HPC while XP has pair programming approaches that support continuous integration and automated builds. Both models in this view are complement to each other. [5,6,7,8,9.10]



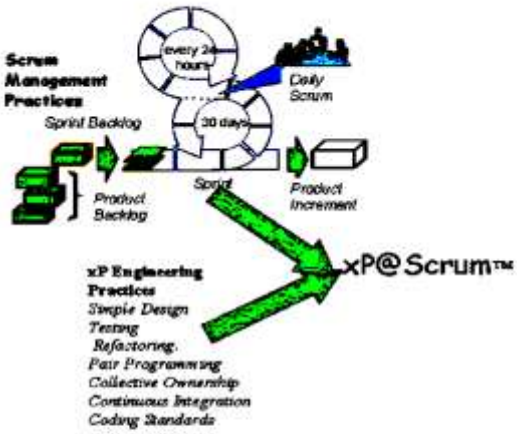


Figure 5 Adopting Scrum-XP Model [28]

The following table indicates the adoption of Scrum and XP that has been carried out in this research. The Scrum-XP model is also used to trace the ability of the system in achieving high performance on different HPC platforms.

Table 3 Scrum-XP in this research

Scrum	XP
1. Iteration of coding of approximately in 30 days. (Sprint planning)	1. Producing simple system design by placing the functionality to be implemented.
2. Prioritized list of tasks and time to complete. (Backlog)	2. First test coding. This step forces the understanding of the interface and expected functionality of a module.
3. Updating list of works and the amount of remaining works to complete. (Sprint backlog)	3. Continuous integration. Automatically check out all codes, build the codes and run the library of tests. This step to ensure the base software is stable and of a high quality.
	4. Refactoring, that allows incremental improvement of design and modules.
	5. Pair programming. Doing all work in small group including data analysis, data model, the incremental of codes, etc.
	6. Customer acceptance test. This is a process to obtain the confirmation that system meets its requirements.

To conduct the empirical analysis of the case study, this research proposes data collection from the sprint release as indicated by the following tables. The data is collected from 3 sprint releases.

All the columns represent cumulative/average data about releases of the case study while all the rows represent data of a particular attribute of the case study.

The first release (row one in the Table 3) was completed in two week time, whereas each of remaining 2 releases took two and four week duration. (row-1)

Table 4 Empirical Analysis of Case Study

ID	Item	Release -1	Release -2	Releas e-3	Total
1	Calendar (week times)	2	2	4	8
2	Number of modules (sprint backlog)	2	1	3	6
3	Total task defined	10	12	9	31
4	Total work effort (h)	200	100	300	600
5	Task allocated actual hours	130	78	240	448
6	Task allocated actual (%)	65%	78%	80%	74%
7	Interface	5	5	5	15
8	Classes	10	15	9	34
9	Test Classes	2	2	2	6
10	Pair Programmin g	80%	80%	80%	80%
11	Customer Satisfaction	78%	80%	80%	79.3 %

The total number of modules indicated at row 2. The total task define was indicated at row 3 with 10 at first release and continue growing on the second and third release by having additional 21 new tasks define. The total work efforts (row 4, 5, 6) have been continually growing especially at third release. The interfaces are constantly growing with same numbers for each release (row 7). The number of classes and test classes are presented at row 8 and 9. Pair programming applied for certain modules that requires intensive coding and was uniform throughout the product development (Parallel feature extraction). And customer satisfaction are achieved at level 79.33% as the system still requires integration with other modules to perform the functions.

8. CONCLUSION

The adaptive agile had supported the development of parallel feature extraction in HPC for most of stages at parallel development life cycle. TShe three main problems that usually met in HPC are able to overcome by adopting the adaptive agile traceable matrix (Table 2 and Table 3) and the combination of two agile practices Scrum-XP (Figure 4). The model validation is confirmed with the customer satisfaction index (Table-4) that suggests continuous improvements to ensure the user level satisfaction. It is also confirmed that the Scrum-XP model is acceptable to be adopted in many HPC applications.

9. REFERENCES

[1] "Manifesto for Agile Software Development", http://agilemanifesto.org

[2] R. Green, T. Mazzuchi and S. Sarkani, "Communication and Quality in Distributed Agile Development: An Empirical Case Study", WASET 2010, pp. 322-328

[3] K. Orr, "Agile Requirements: Opportunity or Oxymoron?", IEEE Software, Vol. 21, No. 3, 2004, pp. 71-73

[4] H. P. Breivold, D. Sundmark, P. Wallin and S. Larsson, "What Does Research Say about Agile and Architecture?",

Sci.Int.(Lahore),25(4),131-137, 2013 ISSN 1013-5316;CODEN: SINTE 8 137

Proceedings of the 2010 Fifth International Conference on Software Engineering Advances, 2010, pp. 32-37.

[5] “What is eXtreme Programming?”, http://www.xprogramming.com/what_is_xp.htm

[6] D. Norton, “What is Scrum?”, http://dotnetjunkies.com/WebLog/darrell.norton/articles/3910.aspx , November 23, 2003.

[7] Extreme Programming official website, (2011). http://www.extremeprogramming.org/map/project.html, visited: 7 August 2013.

[8] M. Rizwan Jameel Qureshi, Empirical Evaluation of the Proposed eXSCRUM Model: Results of a Case Study, IJCSI International Journal of Computer Science Issues, Vol. 8, Issue 3, No. 2, May 2011

[9] C. Juyun, “A hybrid model software development method for large-scale projects”, Journal of Issues in Information Systems, Vol. 10, No. 2, pp. 340-348.

[10] M. Kumar, N. Ajmeri and S. Ghaisas, “Towards knowledge assisted agile requirements evolution”, RSSE '10**:** Proceedings of the 2nd International Workshop on Recommendation Systems for Software Engineering, 2010, pp. 16-20.

[11] J. Baptista, “Agile documentation with uScrum”, Proceedings of the 26th annual ACM international conference on Design of communication, 2008, pp.275-276.

[12] S. Stolberg, “Enabling Agile Testing through Continuous Integration”, Proceedings of the 2009 Agile Conference, 2009, pp.369-274

[13] “What is SCRUM?”, h ttp://www.controlchaos.com/about

[14] K. Schwaber, “SCRUM Development Process”

[15] W. C. Wake, “Scrum Development Process”, h t tp://www.xp123.com , January 25,2004.

[16] Culibrk, Dubravko, Borislav Antic, and Vladimir S. Crnojevic. "Real-time Stable Texture Regions Extraction for Motion-based Object Segmentation." *BMVC*. 2009.

[17] Palomera-Pérez, Miguel A., et al. "Parallel multiscale feature extraction and region growing: application in retinal blood vessel detection." *Information Technology in Biomedicine, IEEE Transactions on* 14.2 (2010): 500-506.

[18] Chung, Yongwha, and Viktor K. Prasanna. "Parallelizing image feature extraction on coarse-grain machines." *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 20.12 (1998): 1389-1394.

[19] Choras, Ryszard S. "Image feature extraction techniques and their applications for CBIR and biometrics systems." *international journal of biology and biomedical engineering* 1.1 (2007): 6-16.

[20] Broggi, Alberto. "Parallel and local feature extraction: a real-time approach to road boundary detection." *Image Processing, IEEE Transactions on* 4.2 (1995): 217-223.

[21] Krishnan, D., Ping Lin, and X. C. Tai. "An efficient operator splitting method for noise removal in images." *Commun. Comput. Phys* 1.5 (2006): 847-858.

[22] Mythili, C., and V. Kavitha. "Efficient Technique for Color Image Noise Reduction." *The research bulletin of Jordan, ACM* 1.11 (2011): 41-44.

[23] Houhou, Nawal, J-P. Thiran, and Xavier Bresson. "Fast texture segmentation model based on the shape operator and active contour." *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*. IEEE, 2008.

[24] Dowd, Kevin, Charles R. Severance, and Michael Kosta Loukides. *High performance computing*. Vol. 2. O'Reilly, 1998.

[25] Fan, Zhe, et al. "GPU cluster for high performance computing." *Proceedings of the 2004 ACM/IEEE conference on Supercomputing*. IEEE Computer Society, 2004.

[26] Huang, Wei, et al. "A case for high performance computing with virtual machines." *Proceedings of the 20th annual international conference on Supercomputing*. ACM, 2006.

[27] Cimic, S. Applying Agility To The Parallel Software Development Lifecycle.

[28] Ken Schwaber and Mike Beedle, Agile Software Development with Scrum (Prentice Hall, 2001, ISBN 0-13-0676

[29] 34-9).