

การเพิ่มประสิทธิภาพของการเขียนโปรแกรมภาษา PHP ร่วมกับโปรแกรม SCILAB

Performance Improvement of Programming a PHP Language in Conjunction with a SCILAB Program

จรรย์นดร บุษวดใช้¹ และ ปิยะ โควินท์พิวีวัฒน์²

¹สถาบันวิจัยและพัฒนา มหาวิทยาลัยราชภัฏนครปฐม
b.jirundon@gmail.com

²โปรแกรมวิชาวิศวกรรมโทรคมนาคม คณะวิทยาศาสตร์และเทคโนโลยี มหาวิทยาลัยราชภัฏนครปฐม
piya@npru.ac.th

บทคัดย่อ

การใช้งานโปรแกรมด้วยภาษา PHP ร่วมกับโปรแกรม SCILAB ที่นำเสนอใน [1] ใช้เวลานานในการประมวลผลและแสดง ผลลัพธ์บนเว็บไซต์ งานวิจัยนี้นำเสนอวิธีการเพิ่มประสิทธิภาพของการเขียนโปรแกรมภาษา PHP ร่วมกับโปรแกรม SCILAB เพื่อให้สามารถทำงานได้เร็วขึ้น โดยได้ออกแบบการทดลองเป็น 4 รูปแบบ คือรูปแบบที่ 1 เป็นการรับ/ส่งข้อมูลผ่านไฟล์ชั่วคราว (แบบเดิมที่ใช้ใน [1]) รูปแบบที่ 2 จะเหมือนกับรูปแบบที่ 1 แต่จะใช้ฟังก์ชันใหม่ของภาษา PHP (สำหรับเวอร์ชัน 5 ขึ้นไป) ในการรับ/ส่งข้อมูล, รูปแบบที่ 3 เป็นการรับ/ส่งข้อมูลโดยตรงไปยังโปรแกรม SCILAB ด้วยวิธีของ Daniel J. Lewis [2] และรูปแบบที่ 4 จะเหมือนกับรูปแบบที่ 3 แต่จะปิดการใช้งานฟังก์ชันที่ไม่จำเป็นในการโหลดโปรแกรม SCILAB ผลการทดลองพบว่ารูปแบบที่ 4 สามารถเพิ่มประสิทธิภาพในการทำงานได้มากที่สุด (ประมาณ 40%) เมื่อเทียบกับรูปแบบที่ 1 โดยยังคงให้ผลลัพธ์ผ่านเว็บไซต์ ถูกต้องทุกขั้นตอน

คำสำคัญ: เว็บไซต์ SCILAB Optimization

Abstract

The use of a PHP Language in conjunction with a SCILAB program presented in [1] normally takes long time to process and display a result on the website. To speed up this technique, we propose a method to increase the performance of programming a PHP Language in conjunction with a SCILAB program. To do so, we design four experiments to test the proposed method. The first method is based on receiving/transmitting a temporary file (as used in [1]). The second method is similar to the first one but using a new function in PHP (valid only for version 5 up) to receive/transmit data. The third method is based on receiving/ transmitting a temporary file directly to a SCILAB program based on the technique proposed by Daniel J. Lewis [2]. Finally, the fourth method is also similar to the third one but we disable some functions that are unnecessary for loading the SCILAB program. Experimental results show that the fourth method can increase the overall performance by approximately 40% if compared with the first method, while maintaining the same outputs in every step via the website.

Keywords: Performance improvement, PHP, SCILAB, website optimization

1. บทนำ

การพัฒนาเว็บไซต์ (หรือการโปรแกรม) ด้วยภาษา PHP [3] ร่วมกับโปรแกรม SCILAB [4] ที่นำเสนอใน [1] จะรับ/ส่งค่าไปประมวลผลที่โปรแกรม SCILAB ผ่านไฟล์ชั่วคราว (Temporary File) ได้ถูกต้องทุกครั้ง แต่จะใช้เวลานานในการแสดงผลที่ออกเว็บไซต์ ดังนั้นงานวิจัยนี้จะศึกษาหาแนวทางในการพัฒนาอัลกอริทึมให้มีสมรรถนะที่ดีที่สุด เพื่อลดเวลาในการประมวลผล อีกทั้งยังช่วยลดภาระการทำงานของเครื่องแม่ข่าย (Server) ที่ติดตั้งโปรแกรม SCILAB อีกด้วย

ในการทดสอบเทคนิคการเพิ่มประสิทธิภาพของการเขียนโปรแกรมภาษา PHP ร่วมกับโปรแกรม SCILAB ผู้วิจัยได้ออกแบบการทดลองออกเป็น 4 รูปแบบ คือ

- รูปแบบที่ 1 คือการรับ/ส่งข้อมูลผ่านไฟล์ชั่วคราว (แบบเดิมที่ใช้ใน [1])
- รูปแบบที่ 2 เหมือนกับรูปแบบที่ 1 แต่จะใช้ฟังก์ชันใหม่ของภาษา PHP (สำหรับ PHP เวอร์ชัน 5 ขึ้นไป)
- รูปแบบที่ 3 คือการรับ/ส่งข้อมูลโดยตรงไปยังโปรแกรม SCILAB ด้วยวิธีของ Daniel J. Lewis [2] (ผู้วิจัยสมมติว่าการสร้างไฟล์ชั่วคราวทำให้การเสียเวลาในการทำงานของทั้งระบบ)

- รูปแบบที่ 4 เหมือนกับรูปแบบที่ 3 แต่จะปิดฟังก์ชันที่ไม่จำเป็นในการโหลดโปรแกรม SCILAB

โดยผู้วิจัยจะจับเวลาที่ใช้ในแต่ละวิธีในตั้งแต่เริ่มส่งข้อมูลจากเว็บไซต์ไปประมวลผลที่โปรแกรม SCILAB ในเครื่องแม่ข่าย จนถึงการส่งผลลัพธ์ที่ได้มาแสดงผลที่เว็บไซต์

2. ประวัติความเป็นมาของโปรแกรม SCILAB และภาษา PHP

โปรแกรม SCILAB เป็นโปรแกรมภาษาขั้นสูงที่ถูกพัฒนาขึ้น โดยความร่วมมือกันระหว่างนักวิจัยจากสถาบัน Institut National De Recherche En Informatique Et En Automatique (INRIA) และ École nationale des ponts et chaussées (ENPC) ประเทศฝรั่งเศส ตั้งแต่ ค.ศ. 1990 โดยมีจุดมุ่งหมายเพื่อใช้ในการคำนวณเชิงตัวเลขและแสดงผลกราฟิกที่ซับซ้อน ดังนั้นโปรแกรม SCILAB จึงเหมาะสำหรับการใช้งานทางด้านวิศวกรรมศาสตร์และวิทยาศาสตร์ นอกจากนี้ โปรแกรม SCILAB ยังเป็นโปรแกรมที่ให้ฟรี (freeware) อนุญาตให้ผู้ใช้สามารถนำไปพัฒนาต่อได้ และมีความสามารถในการทำงานที่ใกล้เคียงกับโปรแกรม MATLAB [5] ซึ่งเป็นโปรแกรมที่นิยมมาก สำหรับผู้ใช้งานทางด้านวิศวกรรมศาสตร์และวิทยาศาสตร์ แต่มีค่าลิขสิทธิ์ซอฟต์แวร์ที่มีราคาแพงมากดังนั้นหน่วยงานต่างๆ ทั้งภาคอุตสาหกรรมและภาคการศึกษา (ทั้งในและนอกประเทศ) ในปัจจุบันได้เริ่มนำโปรแกรม SCILAB มาช่วยในการทำงาน การเรียนการสอน และการทำวิจัย

ภาษา PHP เดิมมาจากคำว่า Personal Home Page แต่ต่อมาเปลี่ยนเป็น PHP Hypertext Preprocessor ซึ่งหมายถึงภาษาคอมพิวเตอร์ที่ใช้ในการพัฒนาเว็บไซต์ในลักษณะแม่ข่าย-ลูกข่าย (Server-Client) โดยสามารถใช้งานได้ฟรี (Open Source) ซึ่งโดยทั่วไปภาษา PHP ใช้สำหรับจัดทำเว็บไซต์แบบพลวัต (Dynamic) และแสดงผลออกมาในรูปแบบเอชทีเอ็มแอล (HTML) โดยมีรากฐานโครงสร้างคำสั่งมาจากภาษาซี ภาษาจาวา และภาษาเพิร์ล (Perl) ดังนั้นภาษา PHP จึงง่ายต่อการเรียนรู้ โดยเป้าหมายหลักของภาษานี้คือ ให้นักพัฒนาเว็บไซต์สามารถเขียนเว็บเพจที่มีความสามารถในการตอบโต้ได้อย่างรวดเร็ว

3. วัตถุประสงค์การวิจัย

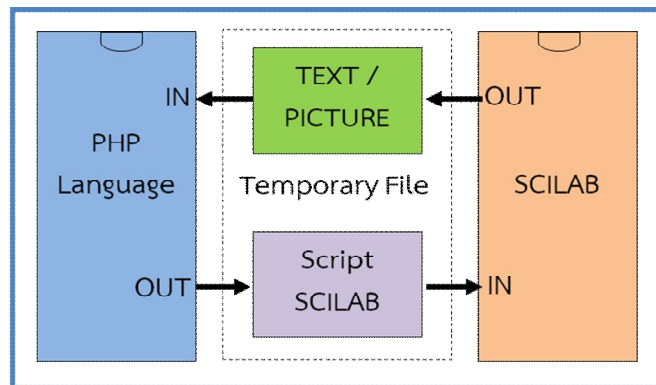
งานวิจัยนี้จะนำเสนอเทคนิคเพื่อเพิ่มประสิทธิภาพการเขียนโปรแกรมด้วยภาษา PHP ร่วมกับโปรแกรม SCILAB ซึ่งจะช่วยให้การแสดงผลที่ (ที่ได้จากการคำนวณของโปรแกรม SCILAB) ออกหน้าเว็บไซต์มีความรวดเร็วมากขึ้น เมื่อเทียบกับเทคนิคแบบเดิมที่ใช้ใน [1]

4. อุปกรณ์ที่ใช้ในการวิจัย

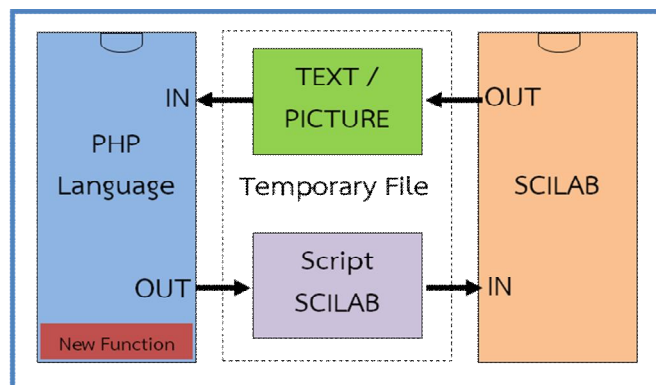
อุปกรณ์ที่ใช้ในการวิจัยนี้แบ่งเป็น 2 ประเภท คือ

- 1) ฮาร์ดแวร์: เครื่องคอมพิวเตอร์ที่ใช้ในการวิจัย โดยมีคุณลักษณะที่สำคัญดังนี้ซีพียู (CPU) แบบ Core2Duo 2.26 GHz, RAM ขนาด 2 GB, ฮาร์ดดิสก์ไดรฟ์ขนาด 250 GB, วีจีเอ (VGA) แบบ 1 GB, และจอแอลซีดีแบบ 1024x768 จดภาพ

2) ซอฟต์แวร์: โปรแกรมที่ใช้ในการวิจัยประกอบด้วย โปรแกรม SCILAB 4.1.2 (สำหรับการคำนวณทางคณิตศาสตร์), โปรแกรม AppServ v.2.5.10 (สำหรับสร้างเว็บไซต์) และโปรแกรม Notepad++ v5.9.8 (เอดิเตอร์สำหรับเขียนโปรแกรม)



ภาพที่ 1 แนวคิดของรูปแบบที่ 1 สำหรับการรับ/ส่งข้อมูลผ่านไฟล์ชั่วคราวแบบเดิม [1]



ภาพที่ 2 แนวคิดของรูปแบบที่ 2 ซึ่งจะเหมือนกับรูปแบบที่ 1 แต่ใช้ฟังก์ชันใหม่ของภาษา PHP เวอร์ชัน 5 ขึ้นไป

5. วิธีการวิจัย

ในการทดสอบเทคนิคที่พัฒนาขึ้นมา ผู้วิจัยได้กำหนดรูปแบบการทดลองออกเป็น 4 รูปแบบ ดังต่อไปนี้

รูปแบบที่ 1 เป็นการรับ/ส่งข้อมูลผ่านไฟล์ชั่วคราวแบบเดิมตามที่นำเสนอใน [1] โดยการส่งข้อมูลจากเว็บไซต์จะถูกเขียนลงไฟล์สคริปต์ของโปรแกรม SCILAB และนำไปประมวลผลในโปรแกรม SCILAB จากนั้นส่งผลลัพธ์ที่ได้ในรูปแบบข้อความหรือรูปภาพมาแสดงผลที่หน้าเว็บไซต์ ดังภาพที่ 1

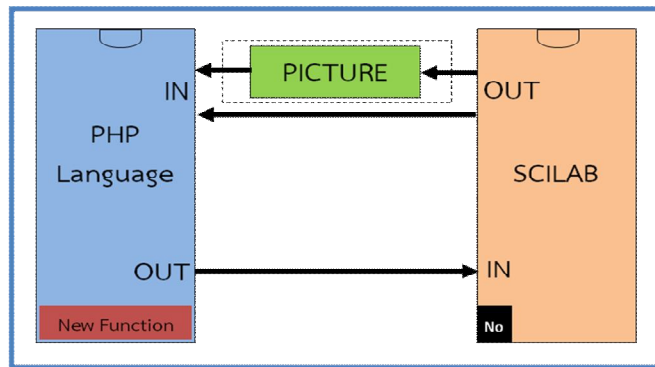
รูปแบบที่ 2 เป็นการรับ/ส่งข้อมูลเหมือนกับรูปแบบที่ 1 เพียงแต่จะใช้ฟังก์ชันใหม่ของภาษา PHP เวอร์ชัน 5 ขึ้นไป ในการการเขียนและอ่านไฟล์ นั่นคือใช้ฟังก์ชัน file_put_contents [6] และ file_get_contents [7] ซึ่งจะนำมาใช้งานแทนฟังก์ชัน fopen fwrite fread และ fclose โดยผู้วิจัยตั้งสมมติฐานว่าฟังก์ชันใหม่นี้จะสามารถทำงานได้รวดเร็วกว่าฟังก์ชันแบบเดิม ตามที่แสดงในภาพที่ 2

รูปแบบที่ 3 เป็นการรับ/ส่งข้อมูลโดยตรงไปยังโปรแกรม SCILAB ด้วยวิธีของ Daniel J. Lewis [2] โดยผู้วิจัยตั้งสมมติฐานว่าการสร้างไฟล์ชั่วคราวเพื่อใช้รับ/ส่งระหว่างโปรแกรม SCILAB และ PHP ทำให้การเสียเวลาในประมวลผลและแสดงผลในเว็บไซต์ ดังนั้นผู้วิจัยจึงได้ทดลองนำวิธีของ Daniel J. Lewis [2] ในการรับ/ส่งข้อมูลโดยตรงไปยังโปรแกรม SCILAB และส่งผลลัพธ์เฉพาะข้อความ (TEXT) โดยตรง (ยังส่งรูปภาพโดยตรงไม่ได้) มาแสดงผลที่หน้าเว็บไซต์ ตามภาพที่ 3

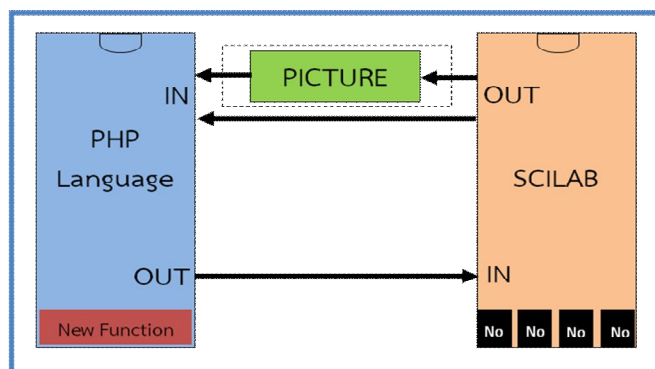
รูปแบบที่ 4 เป็นการรับ/ส่งข้อมูลเหมือนกับรูปแบบที่ 3 แต่จะไปปิดฟังก์ชันที่ไม่จำเป็นในการโหลดโปรแกรม SCILAB เพื่อช่วยทำให้การประมวลผลผลลัพธ์มีความเร็วมากขึ้น โดยมี 4 ขั้นตอนตามภาพที่ 4 ดังนี้

- 1) ไม่ต้องโหลด Banner ของโปรแกรม SCILAB
- 2) ไม่ต้องแสดงหน้าต่างวินโดวส์ของโปรแกรม SCILAB

- 3) ไม่ต้องโหลดข้อมูลผู้ใช้แฟ้มเริ่มต้น (SCIHOME/scilab.ini)
- 4) ไม่ต้องโหลดแฟ้มเริ่มต้นของโปรแกรม SCILAB (scilab.star)



ภาพที่ 3 แนวคิดของรูปแบบที่ 3 สำหรับการรับ/ส่งข้อมูลโดยตรงด้วยวิธีของ Daniel J. Lewis [2] (เฉพาะที่เป็นข้อความ)



ภาพที่ 4 แนวคิดของรูปแบบที่ 4 ซึ่งจะเหมือนกับรูปแบบที่ 3 แต่จะปิดฟังก์ชันที่ไม่จำเป็นในการโหลดโปรแกรม SCILAB

จากนั้นผู้วิจัยได้ทำการพัฒนาเขียนโค้ด (Code) เพื่อใช้ในการทดสอบการทำงานของรูปแบบต่างๆ โดยในที่นี้จะยกตัวอย่างการบวกเลขสองจำนวน และการวาดกราฟ 3 มิติ ด้วยโปรแกรม SCILAB ซึ่งมีรายละเอียดดังต่อไปนี้

การบวกเลขสองจำนวนด้วยโปรแกรม SCILAB

รูปแบบที่ 1 โค้ดการบวกเลขสองจำนวนด้วยโปรแกรม SCILAB

```

1  <?php
2      if(isset($_POST['parameter1']) && isset($_POST['parameter2'])) {
3          $code = "result = " . $_POST['parameter1'] . "+" . $_POST['parameter2'] .
";\nsavematfile('add01.txt','result');\nexit;";
4          $obj=fopen("add1.sce","w+");
5          fwrite($obj, $code);
6          fclose($obj);
7          $path=$_SERVER["SystemRoot"];
8          $com = "$path/scilab-4.1.2/bin/Scilex.exe -f add1.sce";
9          exec($com);
10         $obj2 = fopen("add01.txt","r");
11         if ($obj2) {
12             $_POST['result'] = number_format(fgets($obj2, 4096));
13         }
14         fclose($obj2);
15     }
16  ?>
    
```

รูปแบบที่ 2 โค้ดการบวกเลขสองจำนวนด้วยโปรแกรม SCILAB

```

1 <?php
2   if(isset($_POST['parameter1']) && isset($_POST['parameter2'])) {
3       file_put_contents('add2.sce', 'result = ' . $_POST['parameter1'] . '+' . $_POST['parameter2'] .
";\nsavematfile('add02.txt','result');\nexit;");
4       exec("$_SERVER['SystemRoot'].\\scilab-4.1.2/bin/Scilex.exe -f add2.sce");
5       $_POST['result'] = number_format(file_get_contents("add02.txt"));
6   }
7 ?>

```

รูปแบบที่ 3 โค้ดการบวกเลขสองจำนวนด้วยโปรแกรม SCILAB

```

1 <?php
2   $_POST['result'] = "";
3   if(isset($_POST['parameter1']) && isset($_POST['parameter2'])) {
4       $code = "result = " . $_POST['parameter1'] . "+" . $_POST['parameter2'] . "; disp(result); exit";
5       exec('C:/WINDOWS/scilab-4.1.2/bin/Scilex.exe -nb -e "'.$code.'"', $output);
6       $_POST['result'] = $output[1];
7   }
8 ?>

```

รูปแบบที่ 4 โค้ดการบวกเลขสองจำนวนด้วยโปรแกรม SCILAB

```

1 <?php
2   $_POST['result'] = "";
3   if(isset($_POST['parameter1']) && isset($_POST['parameter2'])) {
4       $code = "result = " . $_POST['parameter1'] . "+" . $_POST['parameter2'] . "; disp(result); exit";
5       exec('C:/WINDOWS/scilab-4.1.2/bin/Scilex.exe -nb -nwni -nouserstartup -ns -e "'.$code.'"', $output);
6       $_POST['result'] = $output[1];
7   }
8 ?>

```

การวาดกราฟ 3 มิติ

รูปแบบที่ 1 โค้ดการวาดกราฟ 3 มิติ

```

1 <?php
2   if(isset($_POST['parameter1'])) {
3       $code = "" . $_POST['parameter1'] . ";\nxs2gif(0,'ex01.gif');\nexit;";
4       $obj=fopen("add1.sce","w+");
5       fwrite($obj, $code);
6       fclose($obj);
7       $path=$_SERVER["SystemRoot"];
8       $com = "$path/scilab-4.1.2/bin/Scilex.exe -f add1.sce";
9       exec($com);
10      echo "<img src='ex01.gif' alt='result' />";
11  }
12 ?>

```

รูปแบบที่ 2 โค้ดการวาดกราฟ 3 มิติ

```

1 <?php
2     if(isset($_POST['parameter1'])) {
3         file_put_contents('add2.sce','".$_POST['parameter1']. ";\nxs2gif(0,'ex02.gif');\nexit;");
4         exec("$_SERVER['SystemRoot'].'/scilab-4.1.2/bin/Scilex.exe -f add2.sce');
5         echo "<img src='ex02.gif' alt='result' />";
6     }
7 ?>

```

รูปแบบที่ 3 โค้ดการวาดกราฟ 3 มิติ

```

1 <?php
2     if(isset($_POST['parameter1'])) {
3         $code = "".$_POST['parameter1'].":xs2gif(0,'ex03.gif');exit;";
4         exec('C:/WINDOWS/scilab-4.1.2/bin/Scilex.exe -nb -e "'.$code.'"');
5         echo "<img src='ex03.gif' alt='result' />";
6     }
7 ?>

```

รูปแบบที่ 4 โค้ดการวาดกราฟ 3 มิติ

```

1 <?php
2     if(isset($_POST['parameter1'])) {
3         $code = "".$_POST['parameter1'].":xs2gif(0,'ex04.gif');exit;";
4         exec('C:/WINDOWS/scilab-4.1.2/bin/Scilex.exe -nb -nwni -nouserstartup -ns -e "'.$code.'"');
5         echo "<img src='ex04.gif' alt='result' />";
6     }
7 ?>

```

6. วิเคราะห์ผลและผลการวิจัย

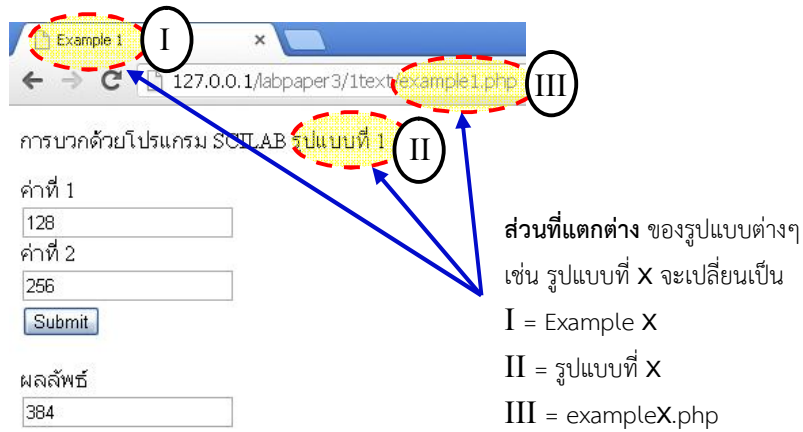
จากนั้นผู้วิจัยได้ทำการจับเวลาตั้งแต่เริ่มต้นจนถึงแสดงผลด้วยฟังก์ชัน microtime [8] โดยทดสอบทั้งหมด 3 ครั้ง และหาค่าเฉลี่ย โดยในที่นี้จะวัดสมรรถนะของระบบจาก 2 การทดลอง ดังนี้

1) สำหรับ “การบวกด้วยโปรแกรม SCILAB” การจับเวลาจะเริ่มตั้งแต่เริ่มต้นกดปุ่ม Submit จนถึงแสดงผล ตามที่แสดงในภาพที่ 5 โดยผลการจับเวลาที่ได้เป็นไปตามตารางที่ 1 ซึ่งจะเห็นได้ว่าเทคนิคที่ใช้ในรูปแบบที่ 4 ใช้เวลาในการทำงานทั้งหมดน้อยกว่ารูปแบบอื่นๆ

2) สำหรับ “การวาดกราฟ 3 มิติด้วยโปรแกรม SCILAB” การจับเวลาจะเริ่มตั้งแต่เริ่มต้นกดปุ่ม Submit จนถึงแสดงผล ตามที่แสดงในภาพที่ 6 โดยผลการจับเวลาที่ได้เป็นไปตามตารางที่ 2 ซึ่งจะเห็นได้ว่าเทคนิคที่ใช้ในรูปแบบที่ 4 ใช้เวลาในการทำงานทั้งหมดน้อยกว่ารูปแบบอื่นๆ

7. สรุปผลการวิจัย

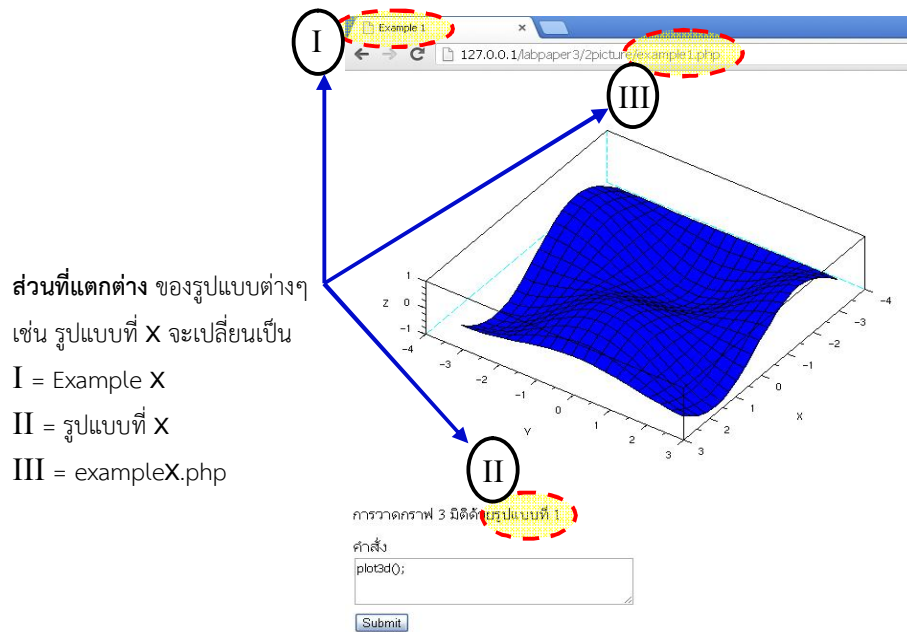
จากการทดลองพบว่ารูปแบบที่ 4 สามารถเพิ่มประสิทธิภาพในการประมวลผลได้ดีที่สุดโดยคิดเป็นเปอร์เซ็นต์จากรูปแบบเดิมประมาณเท่ากับ 40.33 % โดยผลลัพธ์ที่แสดงนั้นถูกต้องทุกครั้งตามการทดลอง ผู้วิจัยจึงมีความมั่นใจว่าการใช้วิธีการเพิ่มประสิทธิภาพเว็บไซต์ภาษา PHP ร่วมกับโปรแกรม SCILAB แบบที่ 4 จะช่วยเพิ่มประสิทธิภาพได้ดีกว่าแบบเดิม



ภาพที่ 5 ตัวอย่างการบวกเลขสองจำนวนด้วยโปรแกรม SCILAB ผ่านเว็บไซต์ เมื่อ $X = \{1, 2, 3, 4\}$

ตารางที่ 1 เวลาที่ใช้ในการบวกเลขสองจำนวนด้วยโปรแกรม SCILAB ผ่านเว็บไซต์

รูปแบบ	ครั้งที่ 1 (วินาที)	ครั้งที่ 2 (วินาที)	ครั้งที่ 3 (วินาที)	ค่าเฉลี่ย (วินาที)
1	0.493756	0.498216	0.521090	0.504354
2	0.489669	0.512839	0.492248	0.498252
3	0.499891	0.476814	0.505074	0.493926
4	0.247293	0.272641	0.254522	0.258152



ภาพที่ 7 ตัวอย่างการวาดกราฟ 3 มิติด้วยโปรแกรม SCILAB ผ่านเว็บไซต์ เมื่อ $X = \{1, 2, 3, 4\}$

ตารางที่ 2 เวลาที่ใช้ในการวาดกราฟ 3 มิติ ด้วยโปรแกรม SCILAB ผ่านเว็บไซต์

รูปแบบ	ครั้งที่ 1 (วินาที)	ครั้งที่ 2 (วินาที)	ครั้งที่ 3 (วินาที)	ค่าเฉลี่ย (วินาที)
1	0.585481	0.577278	0.553122	0.571960
2	0.569492	0.563916	0.563290	0.565566
3	0.574348	0.548335	0.558082	0.560255
4	0.352505	0.367938	0.303453	0.341299

8. กิตติกรรมประกาศ

งานวิจัยชิ้นนี้ได้รับทุนสนับสนุนจากมหาวิทยาลัยราชภัฏนครปฐม “โครงการการพัฒนาสื่อการสอนวิชาคณิตศาสตร์แบบภาพเคลื่อนไหวผ่านเว็บไซต์ โดยใช้โปรแกรม SCILAB ร่วมกับโปรแกรม Flash Animation ปีงบประมาณ 2555”

9. เอกสารอ้างอิง

- [1] จิรันดร บุษวดใช้ และปิยะ โควินท์ทวีวัฒน์, “เทคนิคการใช้งานโปรแกรม SCILAB และภาษา PHP ร่วมกัน,” the 4th NPRU National Academic Conference, Thailand, vol.1, 2012, pp. 227 - 236.
- [2] SCILAB through PHP by Daniel J. Lewis, May1, 2013: www.ibm.com/developerworks/opensource/library/os-php-scilab
- [3] วิกิพีเดีย สารานุกรมเสรี, May 1, 2013: <http://th.wikipedia.org/wiki/ภาษาพีเอชพี>
- [4] SCILAB. May 1, 2013: www.scilab.org
- [5] MATLAB, May1, 2013: <http://www.mathworks.com>
- [6] file-put-contents, May1, 2013: <http://php.net/manual/en/function.file-put-contents.php>
- [7] file-get-contents, May1, 2013: <http://php.net/manual/en/function.file-get-contents.php>
- [8] Script execution time, May, 2013: <http://www.webdesign.org/web-programming/php/script-execution-time.8722.html>
- [9] จิรันดร บุษวดใช้ และปิยะ โควินท์ทวีวัฒน์, “การประยุกต์ใช้โปรแกรม SCILAB ร่วมกับ Flash Animation,” the 4th NPRU National Academic Conference, Thailand, vol.1, 2012, pp. 256 - 264.
- [10] ปิยะ โควินท์ทวีวัฒน์. 2551. คู่มือโปรแกรมภาษา SCILAB สำหรับผู้เริ่มต้น (ฉบับปรับปรุงใหม่). กรุงเทพฯ : เพชรเกษมการพิมพ์ นครปฐม.
- [11] เครื่องมือสำหรับใช้ในการคำนวณ, May 1, 2013: <http://calculator.teenee.com/percentage.html>
- [12] จิรันดร บุษวดใช้ และปิยะ โควินท์ทวีวัฒน์, “การพัฒนาชุดฝึกการเรียนรู้การสอนโดยใช้โปรแกรม VISUAL BASIC ร่วมกับ SCILAB”, NPRU Journal of Science and Technology, Thailand, vol. 3, no. 1, 2010, pp. 113 - 122.