Memorial University of Newfoundland

CMSC6950

Group Project

Growing Degree Days

Author:
Augustine U.O
Obi NixonE.
Gurpreet Singh
Sivachandran Bharathi
Adeoya Adebayo
Boxuan Li

Supervisor:
Dr. Oliver Stueker
Shahrazad Malek
Prof. Ivan Saika-Voivod

June 15, 2017

Abstract

The Growing Degree Day, or GDD, is a heat index that can be used to predict when a crop will reach maturity. Each day's GDD is calculated by subtracting a reference temperature, which varies with plant species, from the daily mean temperature. The reference temperature for a given plant is the temperature below which its development slows or stops. For example, cool season plants, have a reference temperature of 40 degrees fahrenheit while warm season plants, have a reference temperature of 50 degrees fahrenheit. The development of plants depends on the accumulation of heat and since cool season plants have a lower reference temperature, they accumulate GDDs faster than warm season plants.
From http://whyfiles.org

1 Introduction

From wikipedia.org: Growing degree day(s) (GDD), are a measure of heat accumulation used by horticulturists, gardeners, and farmers to predict plant and animal development rates such as the date that a flower will bloom, an insect will emerge from dormancy, or a crop will reach maturity.

Growing degree days take aspects of local weather into account and allow gardeners to predict the plants pace toward maturity.

The base temperature is that temperature below which plant growth is zero. GDs are calculated each day as maximum temperature plus the minimum temperature divided by 2 (or the mean temperature), minus the base temperature. GDUs are accumulated by adding each day's GDs contribution as the season progresses.

2 List of Cities

This are the list of cities we used for this project:

1. St. John's Newfoundland.
2. Ontario Toronto.
3. British Columbia.

2.1 Download daily historical temperature

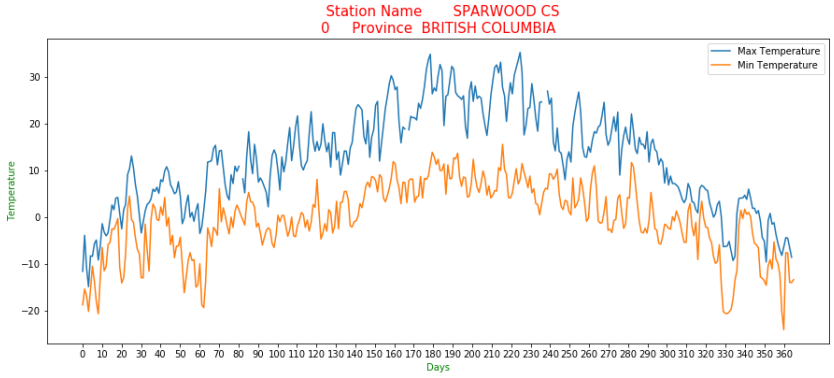
The aim of this section is to download the historical temperature of several cities from the climate.weather database of Canada. The code below extracts the weather informations for the Station Ids, which we will use for the subsequent tasks in this project.

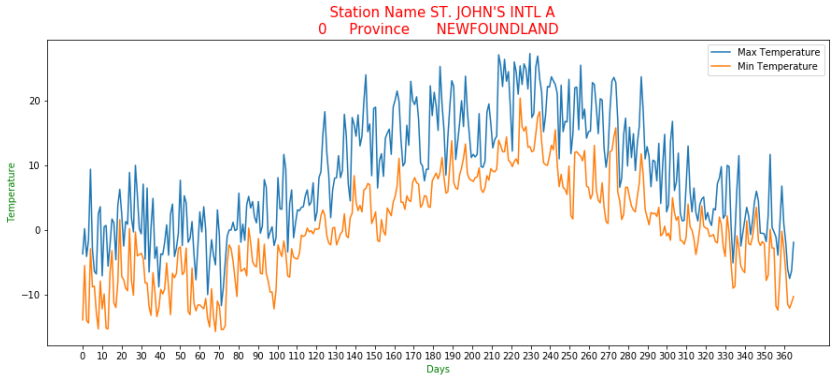
```
def download(stationid, year):


    print("Data for station with id {} for year {}".format(stationid, y


    File_name = "{}_{}_t.csv".format(stationid, year)  #file name for t
    url = "http://climate.weather.gc.ca/climate_data/bulk_data_e.html?fo

    #downloading the file from the URL, and Saving into CSV file
    if(os.path.exists('../Data_csv/'+File_name)): #download only if the
        print("File Exists")
    else:
        print("Downloading in progress ....")
        try:
            urllib.request.urlretrieve(url, '../Data_csv/'+File_name)
        except FileNotFoundError as fnfe:
            print("File not found..")
            print("%s" % fnfe)
            return ""
        except Exception as e:
            print("%s" % e)
            return ""

        print("Download Completed...")
```

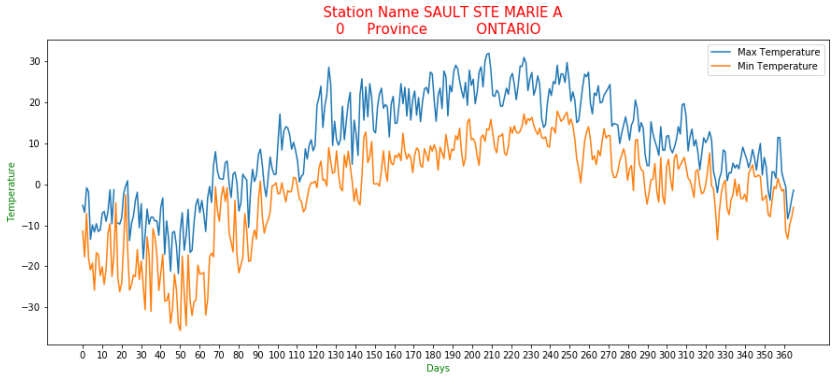
2.2 Annual Cycle of Min/Max Daily Temperatures Plots

In this subsection, we used the downloaded data from the earlier task to create a 3 plots of the selected Canadian cities showing an annual cycle of the minimum or maximum daily temperatures. The min and max temperatures

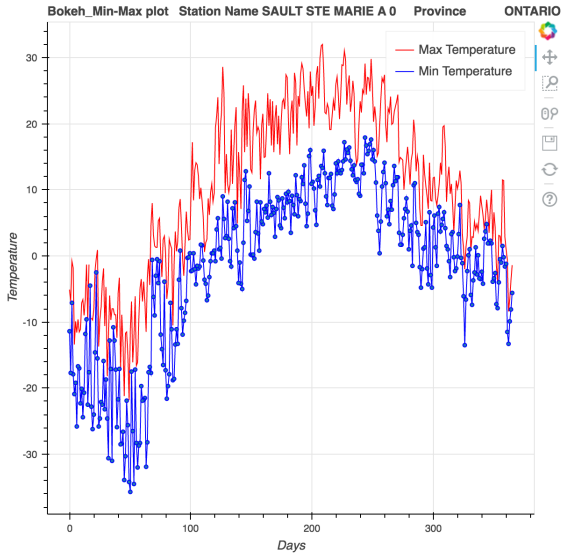
will be used when calculating the GDD in the next task.
When the min-max function is called, the download function gets the information from the station id and year supplied. From the database extracted, we get the min and max temperature which will be used for the plots. Below are the min/max plots for our 3 selected cities:

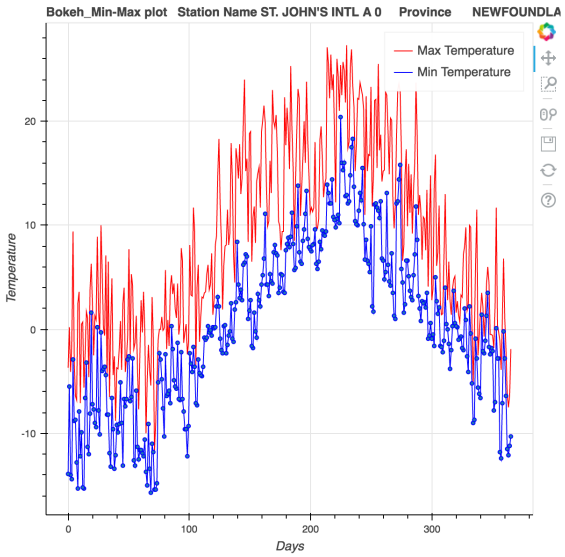


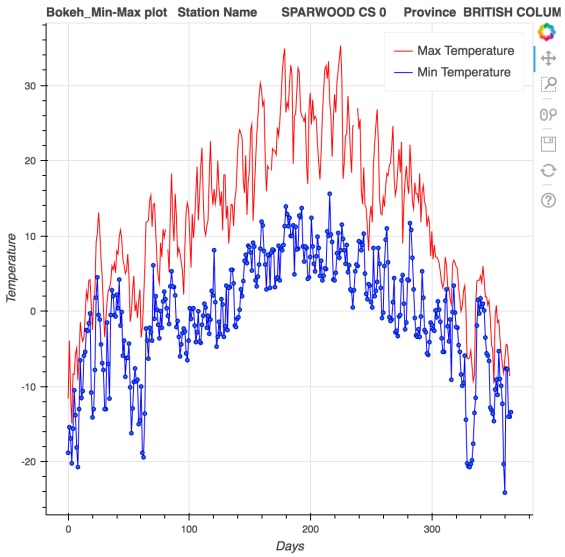




Plotting with Bokeh plot







2.3 Calculating the GDD

To calculate the GDD, we had to use maximum temperature plus the minimum temperature divided by 2 (or the mean temperature), minus the base temperature.
GDD are calculated by taking the integral of warmth above a base temper-

ature, $T_{\text{base}}(usually 10°C)$:
GDD can be calculated using this formulae:

$$GDD = \frac{T_{\max} + T_{\min}}{2} - T_{\text{base}}.$$

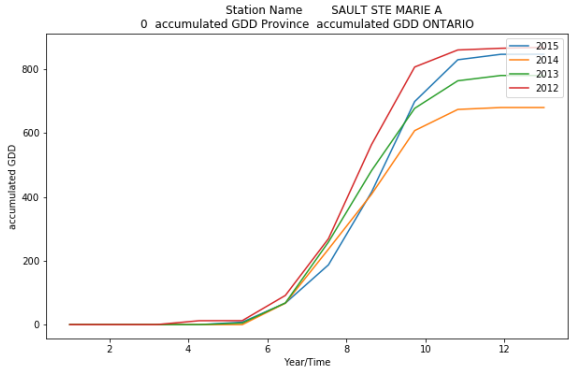
The code below is for calculating GDD:
–Note this is not all the code–

```
import numpy as np
import download_data as DW
import math

def gdd_tot(mean):
    temp = 0.0
    for i in mean:
        if i < 10.0:
            temp = temp + 0.0
        elif not math.isnan(i) :
            temp = temp + (i - 10.0)
    return temp
```

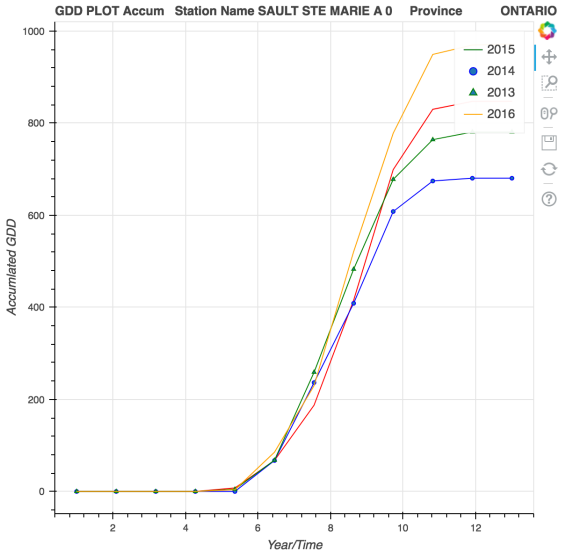
With the gdd-total function, we created a function gdd-cal which will calculate the accumulated gdd for each month using the gdd-total and the gdd-accum will calculate the total of of all the months .
The data calculated will be used to plot the graphs in the next task.

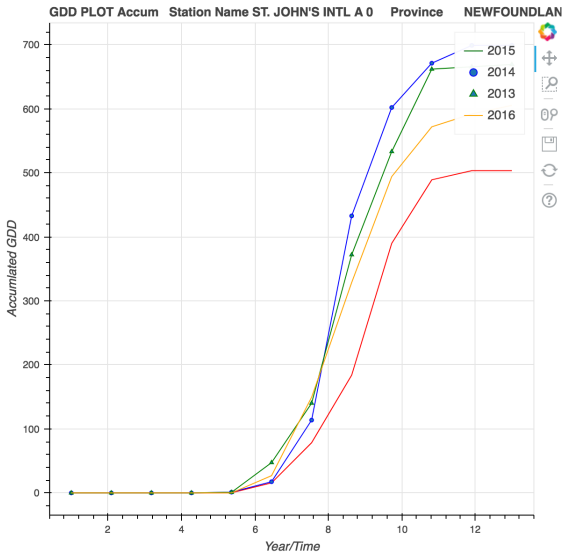
2.4 Plots Showing Accumulated GDD vs Time for Selected Cities

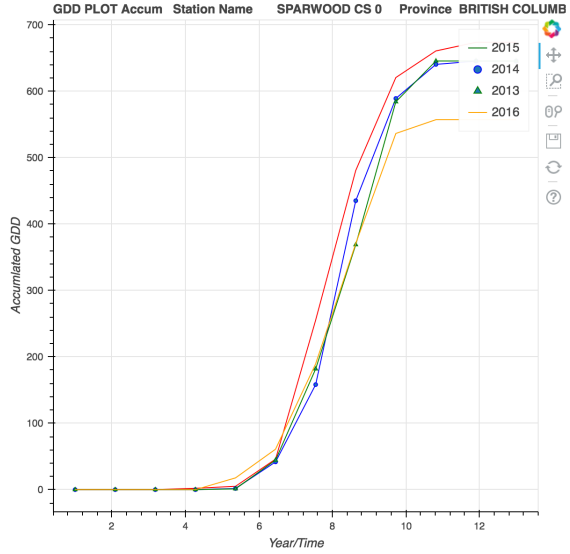
Here we use the results from our calculation above to create plots showing the accumulate GDD vs Time for selected cities.

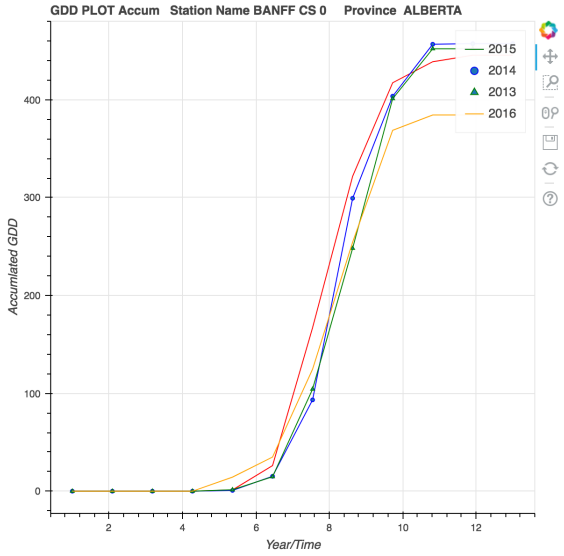

Station Name SAULT STE MARIE A
0 accumulated GDD Province accumulated GDD ONTARIO
2015
2014
2013
2012
accumulated GDD
Year/Time

Bokeh Plots:


GDD PLOT Accum Station Name SAULT STE MARIE A 0 Province ONTARIO
2015
2014
2013
2016
Accumulated GDD
Year/Time

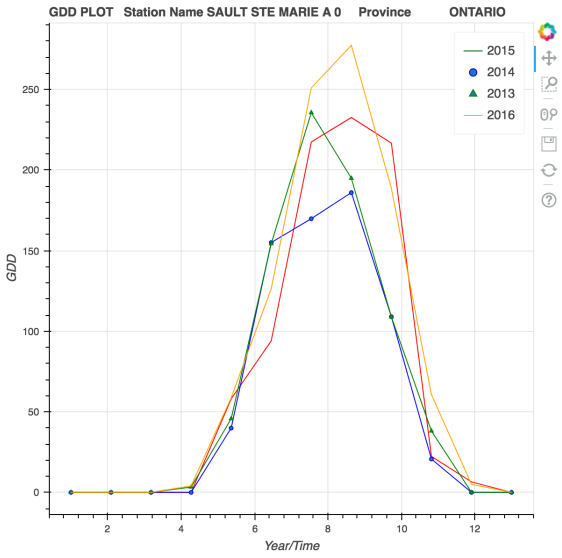

GDD PLOT Accum Station Name ST. JOHN'S INTL A 0 Province NEWFOUNDLAN
2015
2014
2013
2016
Accumlated GDD
Year/Time
0
100
200
300
400
500
600
700
2
4
6
8
10
12

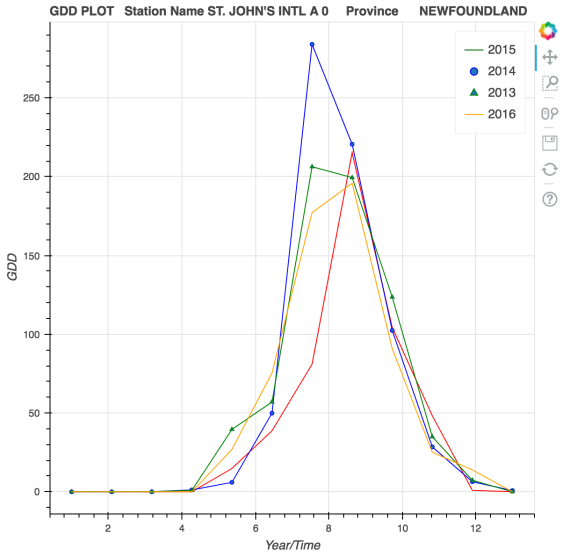

GDD PLOT Accum Station Name SPARWOOD CS 0 Province BRITISH COLUMB
2015
2014
2013
2016
Accumlated GDD
Year/Time
0
100
200
300
400
500
600
700
2
4
6
8
10
12

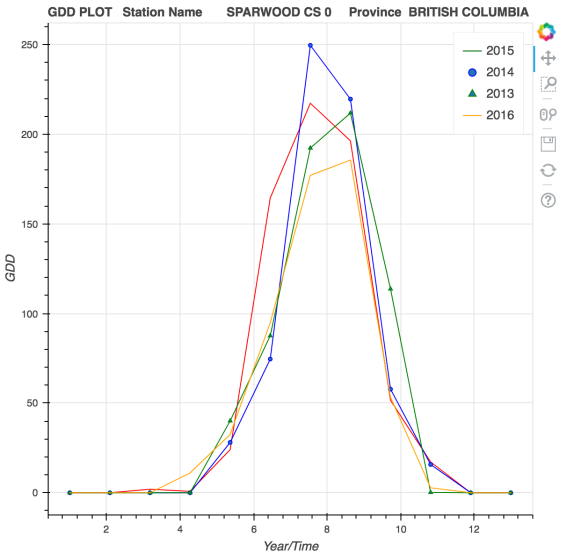


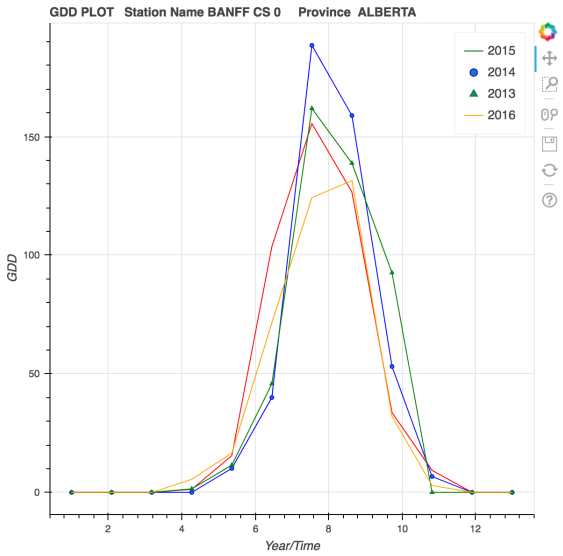
2.5 Secondary Tasks

The plots below for showing GDD:

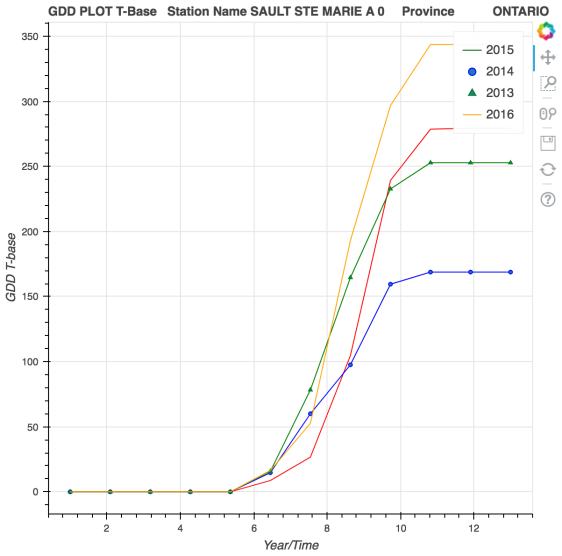


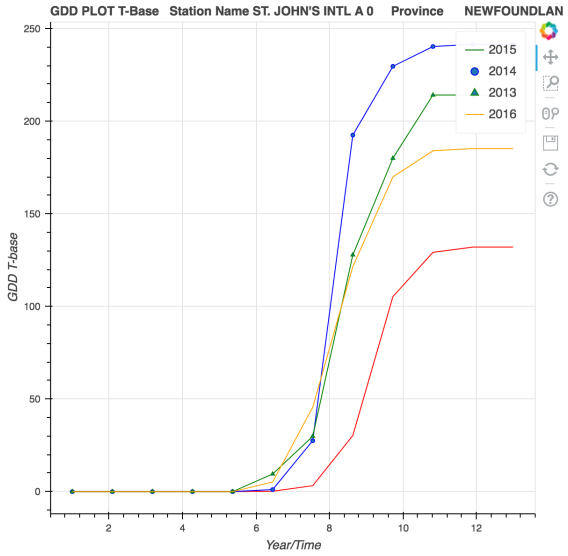

GDD PLOT Station Name ST. JOHN'S INTL A 0 Province NEWFOUNDLAND
2015
2014
2013
2016
GDD
Year/Time

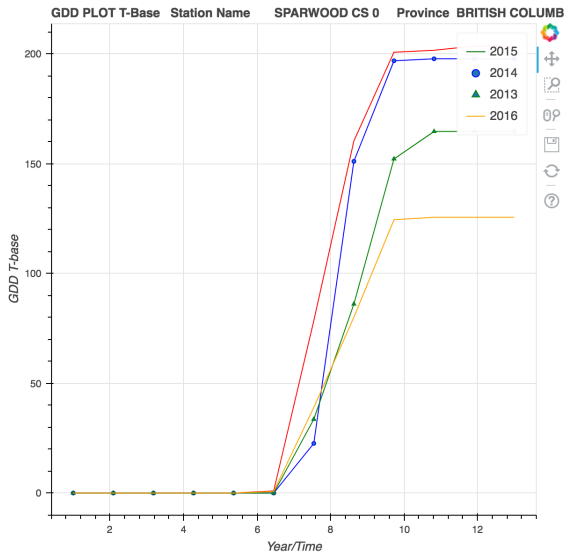

GDD PLOT Station Name SPARWOOD CS 0 Province BRITISH COLUMBIA
2015
2014
2013
2016
GDD
Year/Time

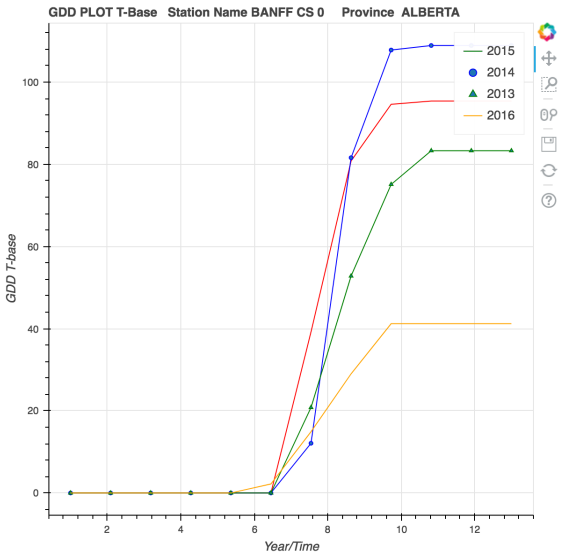


Exploring how GDD calculations depends on the choice of Tbase:




GDD PLOT T-Base Station Name ST. JOHN'S INTL A 0 Province NEWFOUNDLAN
2015
2014
2013
2016
GDD T-base
Year/Time
0
50
100
150
200
250
2
4
6
8
10
12


GDD PLOT T-Base Station Name SPARWOOD CS 0 Province BRITISH COLUMB
2015
2014
2013
2016
GDD T-base
Year/Time
0
50
100
150
200
2
4
6
8
10
12



Time/profile your work flow and estimate the amount of time spent in each step. Explore parallelization to make your work flow execute more quickly.

Profiling the "main" which executes the core task, question 1 secondary task and question 3 of secondary task, shows that the function executing data downloads and computation takes bulk of the execution time. Fortunately, the project required GDD computation for different cities and years. this presents an opportunity for parallelization because computation for one city is independent of another and computation for one year is independent of other years. A parallel implementation of the project based on the number of cities and years is given by the the file "Parallel-main".
The parallel implementation was executed using 1 process, 2 process, 3 process, 4 process, 5 process, 6 process on a computer with 6-cores. The performance of the different number of processes in comparison to the serial implementation (with multiprocessing is given in the table 1 below). The result shows the computation time generally reduced as number of processes increased, although the best performance was obtained using 5 processes. There is usually overhead associated with multiprocessing which could increase with the number of processes running , this may explain why speedup may not perfectly scale up with number of processes used.

3		Test time in seconds					
4			test1	test2	test3	average	speedup
5		serial (no multiprocessing)	8.830	8.911	8.761	8.834	1.000
6		1 Process	8.832	9.043	9.041	8.972	0.985
7		2 Processes	4.987	5.385	4.638	5.003	1.766
8		3 Processes	4.121	4.335	4.300	4.252	2.078
9		4 Processes	3.054	3.245	3.307	3.202	2.759
10		5 Processes	3.184	3.241	3.087	3.171	2.786
11		6 Processes	4.010	3.172	3.074	3.419	2.584