Honda Research Institute EU
Honda Research Institute Europe GmbH
https://www.honda-ri.de/

The Effects of Interaction Frequency in Parallel Cooperative Coevolution

Xiaofen Lu, Stefan Menzel, Ke Tang, Xin Yao

2014

Preprint:

This is a post-peer-review, pre-copyedit version of an article published in The Tenth International Conference on Simulated Evolution And Learning (SEAL 2014). The final authenticated version is available online at: https://doi.org/10.1007/978-3-319-13563-2_8

The Performance Effects of Interaction Frequency in Parallel Cooperative Coevolution

Xiaofen Lu[1], Stefan Menzel[2], Ke Tang[3], and Xin Yao[1,3]

[1] CERCIA, School of Computer Science, University of Birmingham, Edgbaston, Birmingham B15 2TT, UK
[2] Honda Research Institute Europe, 63073 Offenbach/Main, Germany
[3] UBRI, School of Computer Science and Technology, University of Science and Technology of China (USTC), Hefei, Anhui 230027, China
Emails: xxl332@cs.bham.ac.uk, stefan.menzel@honda-ri.de, ketang@ustc.edu.cn, and x.yao@cs.bham.ac.uk

Abstract. Cooperative coevolution (CC) employs a divide-and-conquer paradigm for tackling complex optimization problems. Its performance is influenced by many design decisions. Therefore, to beneficially use it, it is important to acquire some knowledge of the effects of different design settings on the performance of CC. In this paper, we investigate experimentally the performance effects of interaction frequency in parallel CC. The experimental results show that it is overall best for subpopulations to interact with each other as frequently as possible when communication cost is ignored; when communication cost is considered, the best interaction frequency varies from problem to problem and a dynamic change of it is desirable during the optimization process.

Keywords: Cooperative Coevolution, Interaction Frequency

1 Introduction

Cooperative coevolution (CC) is an evolutionary method that tries to solve large-scale problems by problem decomposition. The idea of CC was firstly introduced in genetic algorithm (GA) by Potter and De Jong [11], and later a framework for using the cooperative coevolutionary algorithm (CCEA) was developed [10]. In this framework, the decision variables of an optimization problem are firstly decomposed into several subcomponents, each of which is then evolved in a population. The fitness for each subpopulation individual is assessed by assembling it with representative individuals of the other subpopulations. Experimental results in [11] have shown that this framework can achieve superior performance to traditional GA. Since then, different evolutionary algorithms (EAs) have been successfully extended under this framework due to its generality, such as particle swarm optimization (PSO) ([1], [5]), differential evolution (DE) ([13],[22]), MOEAs [16], and memetic algorithm (MA) [6].

Except combined with different EAs, the CCEA framework has different implementations in different computing environments [3]. In a sequential computing environment, subpopulations in CCEA take turns in evolving. This is called sequential CCEA.

The other is parallel CCEA in which all subpopulations are evolved simultaneously. This is more natural in a parallel computing environment.

As pointed out in [17] and [9], CCEA has many components that can be adjusted, such as how and when to make subpopulations interact with each other. As a result, a practitioner will face a lot of design decisions when applying CCEA to real-world problems. Therefore, it would be useful to generate the knowledge about the performance effects of different choices of these components in CCEA. In literature, various studies have been carried out along the directions of how to select representative individuals, how many representatives to select, and how to calculate the fitness of each subpopulation individual based on these representatives for both parallel CCEA ([2], [7],[19]) and sequential CCEA ([18], [8]). For when to make subpopulations interact with each other, the performance effects of interaction frequency was systematically studied for sequential CCEA [9] but little has been done for parallel CCEA in literature.

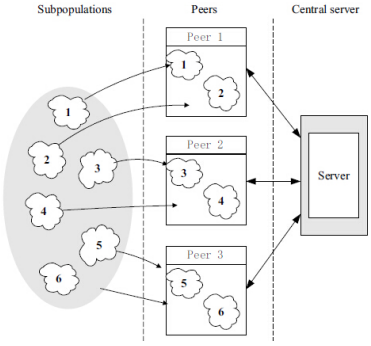


Fig. 1. The Model of Parallel CCEA, taken from [16]

In this paper, we focus on studying the performance effects of communication frequency in parallel CCEA. Different from sequential CCEA, communication cost exists in parallel CCEA. Consider one model of parallel CCEA given in Fig. 1, in which subpopulations are decomposed into several groups and these groups will be distributed over several peer computers. The interaction among subpopulations is indirectly achieved through the exchange of information between the peers and a central server. In such a case, communication overhead can not be ignored especially for distributed networks with limited communication speeds. As communication cost is involved, the performance influence of communication frequency in parallel CCEA would be more complicated than in sequential CCEA.

In CCEA, one critical step is problem decomposition. In [11], each dimension was considered as a subcomponent and evolved in a population. The experimental results have shown that this decomposition method loses its efficiency on non-separable problems. Since then, various studies ([20], [21], [12], [4], [15]) have been carried out towards grouping interdependent variables into the same subcomponent during the search process. In this paper, we consider fixed problem decomposition. In an extreme case,

when the subcomponents are independent, they do not need to communicate with each other; when the subcomponents are interdependent, intuitively, it is better for them to interact with each other. This means that the best communication frequency depends on the correlation degree of subcomponents.

Considering this, our first and second hypotheses are that the performance effects of interaction frequency depends on communication cost and correlation degree of subcomponents, respectively. Besides, our another hypothesis is that the performance effects relate to the evolution speed of the EA. To confirm our hypotheses, we observe the change of the best communication frequency using different EAs, different communication costs, and test functions with different subcomponent correlation degrees. Through this, we also try to answer the questions as follows.

1. Is there an optimal setting for the interaction frequency of parallel CCEA for a particular class of problems?
2. How does the best interaction frequency vary in different EAs?
3. How does the best communication frequency relate to communication cost?

The remaining part of this paper is organized as follows. Section 2 will detail the experiment methodology and experimental setup. Section 3 will give the experimental results and analysis. In Section 4, answers to the above-mentioned questions will be given as well as the directions of future work.

2 Experiment Methodology

2.1 Correlation of Subcomponents

The correlation degree of two subcomponents in this study is calculated by averaging the correlation degree of variables between them, which is calculated based on the statistical interdependence learning model proposed in [15]. That is:

- Suppose we have decision vectors $\overrightarrow{\alpha} = (..., x_i, ..., x_j, ...)$, $\overrightarrow{\beta} = (..., x_i^{'}, ..., x_j, ...)$, and they satisfy that $f(\overrightarrow{\alpha}) \leq f(\overrightarrow{\beta})$ (f is the function to optimize). If the value of x_j is changed to $x_j^{'}$, resulting in $\overrightarrow{\alpha}' = (..., x_i, ..., x_j^{'}, ...)$, $\overrightarrow{\beta}' = (..., x_i^{'}, ..., x_j^{'}, ...)$, and $f(\overrightarrow{\alpha}') > f(\overrightarrow{\beta}')$, then it is called variable x_i is affected by variable x_j under context vector $\overrightarrow{c} = (..., x_{i-1}, -, x_{i+1}..., x_{j-1}, -, x_{j+1}...)$.
- The extent to which the variable x_i can be affected by x_j depends on the probability of the inequality change, i.e., $P\{f(\overrightarrow{\alpha}') > f(\overrightarrow{\beta}')\}$, which can be estimated by selecting a number of $\overrightarrow{c}$ and checking the affect of x_j on x_i under each sample $\overrightarrow{c}$.
- The correlation degree of two variables is calculated by averaging their separate probabilities of the inequality change.

Note that it has been validated in [9] that the performance effects of interaction frequency are dependent on a problem property called *best-response curves*. In this study, we instead use the correlation degree of subcomponents to depict the problem property. The reason is that the later could be much easier to estimate than the former if the information is possibly used.

2.2 The Domains

We start by using three families of test functions, each with a dimension of 2. One is taken from [9], defined as follows:

$$BR_n^{\alpha}(x,y) = \begin{cases} 2y + \frac{\alpha-3}{2\alpha}(x-n), & \text{if } \alpha y < x + (\alpha-1)n; \\ 2x + \frac{\alpha-3}{2\alpha}(y-n), & \text{if } y > x + (\alpha-1)n; \\ n + \frac{x+y}{2}, & \text{otherwise} \end{cases} \quad (1)$$

where $n \in \mathbf{N}, \alpha \in [0,1], x, y \in [0,n]$. The optimum for this function is $BR_n^{\alpha}(n,n) = 2n$. In this paper, n is set to 10.

The other two are the shifted rotated sphere function and shifted rotated rastrigin's function [14], which are typical unimodal and multimodal function, respectively, defined as:

$$F_{sphere}(x) = \sum_{i=1}^{D} z_i^2, \mathbf{z} = (\mathbf{x} - \mathbf{o}) * \mathbf{M}, \mathbf{x} = [x_1, x_2, ..., x_D], \mathbf{x} \in [-100, 100]^D \quad (2)$$

$$F_{sphere}(x) = \sum_{i=1}^{D} (z_i^2 - 10cos(2\pi z_i) + 10), \mathbf{z} = (\mathbf{x} - \mathbf{o}) * \mathbf{M}, \\ \mathbf{x} = [x_1, x_2, ..., x_D], \mathbf{x} \in [-5, 5]^D \quad (3)$$

where D is the number of dimensions, $\mathbf{o} = [o_1, o_2, ..., o_D]$ is the shifted global optimum. In this paper, D is set to 2.

We chose these three functions because the correlation degree of x (or x_1) and y (or x_2) can be changed by changing α (or $\mathbf{M}$). For $BR_n^{\alpha}(x,y)$, when increasing α from 0 to 1, the correlation degree of x and y increases monotonically from 0. For $F_{sphere}(x)$ and $F_{rastrigin}(x)$, assume $\mathbf{M} = \begin{bmatrix} m_{11} & m_{12} \\ m_{21} & m_{22} \end{bmatrix}$ and the initial $\mathbf{M}$ is an identity matrix. By increasing both m_{12} and m_{21} to the same value from 0 to 1, the correlation degree of x_1 and x_2 monotonically increases from 0.

2.3 The CCEA Configuration

We use a two-population CCEA as the basic setup, in which one population is used to evolve x (or x_1) and the other is to evolve y (or x_2). For each population, we run a non-overlapping generational EA. The parameter setting for the used EA in our experiments is given in Table 1.

At first, an initial population is generated for each subcomponent. The population member of each population is evaluated by assembling it with a randomly chosen member of the other population. Then, each population is simultaneously evolved with the above-mentioned EA. The fitness of a population member is evaluated by combining it with the current best member of the other population. During the search process, they communicate with each other their current best solution every constant number of generations.

Table 1. Parameter Setting for EA

Population size (NP)	20
Mutation Probability (P_{mut})	0.9
Evolutionary Operators	Gaussian mutation, elitism of size 1, tournament selection of size 2

2.4 Experimental Setup

In our experiments, we vary the interaction frequency and observe the changes in the performance of CCEA. To confirm the hypotheses, we observe the change of the best interaction frequency by using the above-mentioned test functions with different $alpha$ and $\mathbf{M}$ values, the EA with different Gaussian mutation sigma, and different communication costs, respectively. We set the number of maximum fitness evaluations constant across experiments, and independently run each experiment for 100 times, and then use Wilcoxon rank-sum test at a 0.05 significance level to make comparisons.

We start from the extreme scenario in which communication cost equals 0. In this scenario, we study the relationship between the best communication frequency and each of correlation degree of subcomponents and the mutation step size. Then, we consider the scenario in which communication has a cost and study how the best communication frequency changes as each of correlation degree of subcomponents, the mutation step size, and communication cost changes.

3 Experimental Studies: Communication without Cost

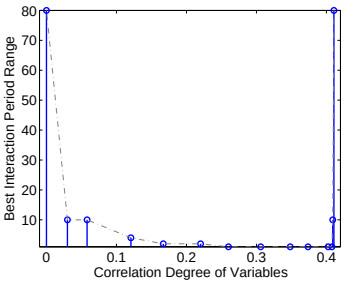
3.1 Different Correlation Degrees of Variables

In this study, the maximum number of fitness evaluations was set to 3220, and thus the maximum number of generations is 80 ($20 + 80 * (2 * 20) = 3220$). The Gaussian mutation sigma was set to 0.2, 1.0, and 0.2 for $BR_n^\alpha(x, y)$, $F_{sphere}(x)$, and $F_{rastrigin}(x)$, respectively. We varied the communication period from 1 to 80 generations and used Wilcoxon rank-sum test to make comparisons. Here, communication period means every how many generations two populations interact with each other. Table 2 summarizes the interval of best communication periods for each test function, which means communication periods in this interval performs the same best according to statistical test.

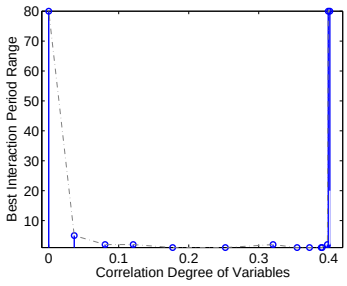
It can be seen from Table 2 that communication frequency indeed has an effect on the performance of parallel CCEA as not all values in [1,80] performed the same best, and thus worths studying . For each family of test functions, the interval of the best communication period varies as the correlation degree of variables varies, and the change tendency is similar. That is, when the variables are independent, all interaction periods performs the same; as the interdependence degree of subcomponents increases, the algorithm more and more prefers short interaction periods; when the correlation degree of variable is high enough, the upper bound of the best interaction period interval increases as correlation degree increases. This can substantiate our first hypothesis.

Table 2. The Interval of Best Interaction Periods (denoted as I_{bp}) for $BR_n^{\alpha}(x,y)$ with Different α Values, $F_{sphere}(x)$ and $F_{rastrigin}(x)$ with Different m_{12}/m_{21} Values, respectively (the column of 'corr' is the estimated correlation degree of x (or x_1) and y (or x_2))

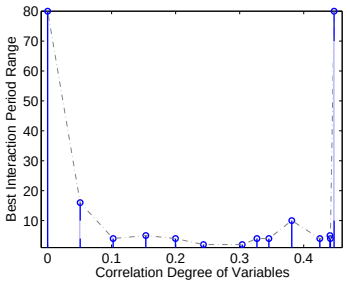
$BR_n^{\alpha}(x,y)$			$F_{sphere}(x)$			$F_{rastrigin}(x)$		
α	$corr$	I_{bp}	m_{12}/m_{21}	$corr$	I_{bp}	m_{12}/m_{21}	$corr$	I_{bp}
0.0	0.0	[1,80]	0.0	0.0	[1,80]	0.0	0.0	[1,80]
0.05	0.0299	[1,10]	0.05	0.0364	[1,5]	0.01	0.0509	[1,4]$\bigcup$[10,16]
0.10	0.058	[1,10]	0.10	0.0805	[1,2]	0.02	0.1024	[1,4]
0.20	0.1206	[1,4]	0.15	0.1207	[1,2]	0.03	0.1534	[1,5]
0.30	0.1669	[1,2]	0.20	0.1770	[1]	0.04	0.1997	[1,4]
0.40	0.2202	[1,2]	0.30	0.2523	[1]	0.05	0.2434	[1,2]
0.50	0.2601	[1]	0.40	0.3205	[1,2]	0.068	0.3040	[1,2]
0.60	0.3061	[1]	0.50	0.3549	[1]	0.08	0.3270	[1,4]
0.70	0.3483	[1]	0.60	0.3726	[1]	0.10	0.3456	[1,4]
0.80	0.3737	[1]	0.70	0.3907	[1]	0.20	0.3813	[1,10]
0.90	0.4025	[1]	0.80	0.3893	[1]	0.40	0.4251	[1,4]
0.95	0.4078	[1]	0.90	0.3981	[1,2]	0.60	0.4410	[1,5]
0.993	0.4090	[1,10]	0.95	0.3994	[1,80]	0.80	0.4416	[1,4]
1.0	0.4107	[1,80]	1.0	0.4017	[1,2]$\bigcup$[20,80]	1.0	0.4474	[1,10]$\bigcup$[70,80]



(a) $BR_n^{\alpha}(x,y)$



(b) F_{sphere}



(c) $F_{rastrigin}$

Fig. 2. The Curves of Best Interaction Period Range versus Correlation Degree of Variables

To get a clearer picture of how the best interaction period interval changes as the correlation of variables changes, the curves of the best communication period interval versus the correlation degree of variables for $BR_n^\alpha(x,y)$, $F_{sphere}(x)$, and $F_{rastrigin}(x)$ are plotted in Fig. 2. It can be seen from Fig. 2 that the upper bound of the interval and the correlation of subcomponents are non-linearly and complicatedly related. Take a closer look at Fig. 2, this non-linear relationship is much higher when the correlation degree is at about 0.4 than when the correlation degree is less than 0.2. When the correlation degree of subcomponents falls in the range between 0.2 to 0.35, it is highest required that communication happens frequently. Moreover, the lower bound of the communication interval is always 1, which means that overall every-generation communication is best when communication cost can be neglected.

3.2 Different Mutation Step Sizes

In this study, we varied Gaussian mutation sigma and recorded the best interaction period interval. Table 3 gives the best interaction frequency interval over $BR_n^\alpha(x,y)$ for three different sigma values (0.1, 0.2, and 0.5).

When the mutation sigma changes from 0.1 to 0.2, we cannot observe an obvious trend of the change of the best communication period interval from Table 3. But, when the sigma changes from 0.1 (or 0.2) to 0.5, it can be seen that the range of the best interaction period decreases while short periods are still preferred.

Table 3. The Interval of Best Interaction Periods (denoted as I_{bp}) for $BR_n^\alpha(x,y)$ with Different Gaussian Mutation Sigma Values

$BR_n^\alpha(x,y)$		I_{bp}		
α	$corr$	$\sigma = 0.1$	$\sigma = 0.2$	$\sigma = 0.5$
0.0	0.0	[1,80]	[1,80]	[1,80]
0.05	0.0299	[1,10]	[1,10]	[1,8]
0.10	0.0580	[1,2]⋃[5]	[1,10]	[1,5]
0.20	0.1206	[1,5]	[1,4]	[1,4]
0.30	0.1669	[1,2]	[1,2]	[1,4]
0.40	0.2202	[1,2]	[1,2]	[1,2]
0.50	0.2601	[1]	[1]	[1]
0.60	0.3061	[1]	[1]	[1]
0.70	0.3483	[1]	[1]	[1]
0.80	0.3737	[1]	[1]	[1]
0.90	0.4025	[1]	[1]	[1]
0.95	0.4078	[1]	[1]	[1]
0.993	0.4090	[1,4]⋃[16,80]	[1,10]	[1,2]
1.0	0.4107	[1,80]	[1,80]	[1,4]

In general, if communication cost can be ignored, it is best for subpopulations to communicate every generation. However, it should be noted here in previous experiments we did not reevaluate each current subpopulation after the communication

happens. Considering that in some implementation of parallel CCEA (e.g. Differential Evolution (DE) is chosen as the optimization algorithm) the current population needs to be reevaluated when information of the best individual in the other subpopulation changes, we further did experiments with reevaluation over $BR_n^\alpha(x,y)$, $F_{sphere}(x)$, and $F_{rastrigin}(x)$, and the results are summarized in Table 4. It can be seen from Table 4 that the lower bound of the best interaction interval changes to 2 or 3 over some cases of $F_{sphere}(x)$. The reason why every-generation communication did not perform best in these cases may be that reevaluation would not help improving the best solution but waste more fitness evaluations if the information does not change too much. Thus, if CCEA is implemented with reevaluation, it should be checked whether the reevaluation is needed every exchange of information.

Table 4. The Interval of Best Interaction Periods (I_{bp}) with Reevaluation for $BR_n^\alpha(x,y)$ with Different α Values, $F_{sphere}(x)$ and $F_{rastrigin}(x)$ with Different m_{12}/m_{21} Values

$BR_n^\alpha(x,y)$			$F_{sphere}(x)$			$F_{rastrigin}(x)$		
α	$corr$	I_{bp}	m_{12}/m_{21}	$corr$	I_{bp}	m_{12}/m_{21}	$corr$	I_{bp}
0.0	0.0	[1,79]	0.0	0.0	[2,79]	0.0	0.0	[1]$\bigcup$[7,79]
0.05	0.0299	[1,9]	0.05	0.0364	[1,4]	0.01	0.0509	[1,19]
0.10	0.0580	[1,4]	0.10	0.0805	[1,3]	0.02	0.1024	[1,7]
0.20	0.1206	[1,3]	0.15	0.1207	[1,3]	0.03	0.1534	[1,9]
0.30	0.1669	[1,3]	0.20	0.1770	[1,3]	0.04	0.1997	[1,7]
0.40	0.2202	[1,2]	0.30	0.2523	[2,3]	0.05	0.2434	[1,7]
0.50	0.2601	[1,2]	0.40	0.3205	[3]	0.068	0.3040	[1,4]
0.60	0.3061	[1]	0.50	0.3549	[2]	0.08	0.3270	[1,4]
0.70	0.3483	[1]	0.60	0.3726	[2]	0.10	0.3456	[1,4]
0.80	0.3737	[1]	0.70	0.3907	[2]	0.20	0.3813	[1,19]
0.90	0.4025	[1]	0.80	0.3893	[1]	0.40	0.4251	[1,3]
0.95	0.4078	[1]	0.90	0.3981	[1,79]	0.60	0.4410	[1,7]
0.993	0.4090	[1,9]	0.95	0.3994	[1,79]	0.80	0.4416	[1,7]
1.0	0.4107	[1,79]	1.0	0.4017	[1]$\bigcup$[39,79]	1.0	0.4474	[1,15]$\bigcup$[69,79]

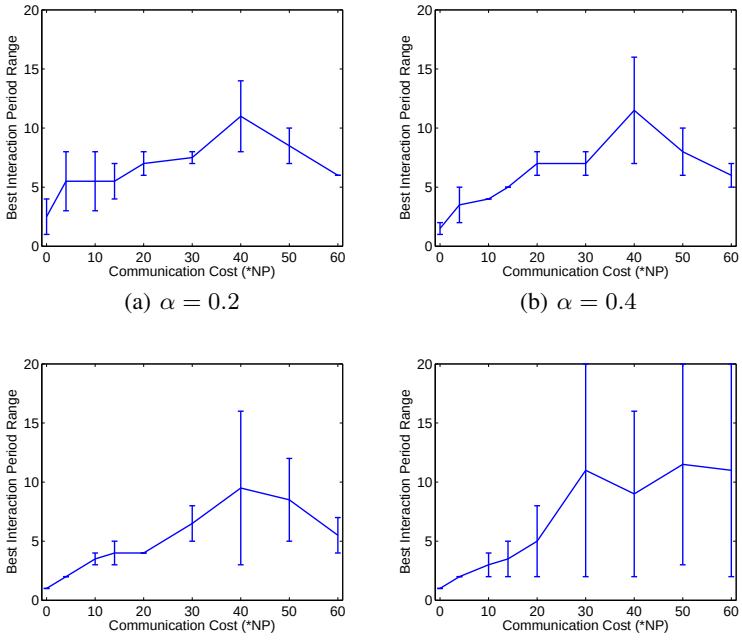
4 Experimental Studies: Communication with Cost

4.1 Different Communication Costs

In this study, we assume communication cost as a fixed number of fitness evaluations as fitness evaluation denotes the time elapse in the CCEA framework. We used 7 cases of $BR_n^\alpha(x,y)$. The maximum number of of fitness evaluations was set to 3000. The interaction period varies from 1 to 20 generations while the communication cost varies from $0 * NP$ to $60 * NP$ (NP is the population size) fitness evaluations. The interval of the best interaction periods for each test function and communication cost is given in Table 5.

Table 5. The Interval of Best Interaction Periods (I_{bp}) over $BR_n^{\alpha}(x, y)$ with Different Communication Costs

	I_{bp}						
cost	$\alpha = 0.0$	0.2	0.4	0.6	0.8	0.99	1.0
	$corr = 0.0$	0.1206	0.2202	0.3061	0.3737	0.4090	0.4107
0	[1,20]	[1,4]	[1,2]	[1]	[1]	[1,10]	[1,20]
$4 * NP$	[12,20]	[3,8]	[2,5]	[2]	[2]	[1,20]	[1,20]
$10 * NP$	[10,20]	[3,8]	[4]	[3,4]	[2,4]	[1,20]	[1,20]
$14 * NP$	[10,20]	[4,7]	[5]	[3,5]	[2,5]	[1,20]	[1,20]
$20 * NP$	[14,20]	[6,8]	[6,8]	[4]	[2,8]	[1,20]	[1,20]
$30 * NP$	[14,20]	[7,8]	[6,8]	[5,8]	[2,20]	[1,20]	[1,20]
$40 * NP$	[16,20]	[8,14]	[7,16]	[3,16]	[2,16]	[1,20]	[1,20]
$50 * NP$	[16,20]	[7,10]	[6,10]	[5,12]	[3,20]	[1,20]	[1,20]
$60 * NP$	[18,20]	[6]	[5,7]	[4,7]	[2,20]	[1,20]	[1,20]



(a) $\alpha = 0.2$ (b) $\alpha = 0.4$

(c) $\alpha = 0.6$ (d) $\alpha = 0.8$

Fig. 3. The Curves of Best Interaction Period Range versus Different Communication Costs

It can be seen from Table 5 that every-generation communication no longer performed best when communication exists. For uncorrelated subcomponents, the largest communication period performed best. For strongly-related subcomponents, all communication periods performed the same. For other functions, the best communication periods are different for different communication costs. This confirms our second hypothesis. Moreover, when the communication cost is the same, the best communication periods are different for different correlation degrees of subcomponents. This substantiates our first hypothesis.

To get a picture of how best interaction period changes as the communication cost changes, the curves of the best communication period interval versus communication cost for $\alpha = 0.2$, $\alpha = 0.4$, $\alpha = 0.6$, and $\alpha = 0.8$ are plotted in Fig. 3. It can be seen that there is a non-linear relationship between the interval of best interaction period and communication cost, and the relationship varies for different correlation degrees of subcomponents. Overall, when communication cost becomes larger, the best interaction period is prone to larger values (note that the decreasing part of the curve as the communication cost increases is because we set the maximum number of fitness evaluations fixed across all communication costs).

4.2 Different Mutation Step Sizes

In this experiment, we change Gaussian mutation sigma and recorded the best interaction period interval over 10 cases of $BR_n^\alpha(x, y)$ with fixed communication cost ($4 * NP$ fitness evaluations). We calculate the number of fitness evaluations to achieve the function value of 19.99 for each interaction period in [1,20], and obtain the best interaction period according to the Wilcoxon rank-sum test. Table 6 shows the interval of best interaction period for three different sigma values (0.05, 0.1, and 0.2).

Table 6. The Interval of Best Interaction Periods (denoted as I_{bp}) for Different Gaussian Mutation Sigma Values over $BR_n^\alpha(x, y)$ with Communication Cost ($4 * NP$)

$BR_n^\alpha(x, y)$		I_{bp}		
$alpha$	$corr$	$\sigma = 0.05$	$\sigma = 0.1$	$\sigma = 0.2$
0.0	0.0	[7,20]	[6,20]	[7,20]
0.05	0.0299	[5, 15]	[4,9]	[3,8]
0.10	0.058	[5,13]	[3,7]	[3,4]
0.20	0.1206	[4,9]	[3,7]	[2,4]
0.30	0.1669	[3,8]	[2,5]	[2,3]
0.40	0.2202	[3,7]	[2,4]	[2,3]
0.50	0.2601	[3,6]	[2,4]	[2]
0.60	0.3061	[2,5]	[2,3]	[2]
0.70	0.3483	[2,4]	[2,3]	[2]
0.80	0.3737	[2,3]	[1,2]	[2]

By comparing the best interaction period interval for 0.05, 0.1, and 0.2, it can be seen that when the mutation step size decreases, the best interaction period will prefer

larger values. This confirms our third hypothesis. Furthermore, this indicates that a different interaction period needs to be set if the evolution speed changes during the search process.

5 Conclusion and Future Work

In this paper, we studied the communication frequency effects on the performance of parallel CCEA. We showed that the best interaction frequency depends on the correlation degree of subcomponents, the evolution speed of the optimization algorithm as well as the communication cost.

As an answer to question 1, for problems that communication cost can be comparatively ignored, it is best to communicate as frequently as possible. For problems that consider communication cost, the best interaction period is hard to set, it is not only related to the correlation of variables and communication cost but also to the evolution speed of the optimization algorithm. But, for separable variables, it is best to set the communication frequency as low as possible.

As an answer to question 2, if the evolution speed of optimization algorithm is quick, high communication frequency is needed. Actually, during the evolutionary process of a CCEA, the evolution speed would change. It would become slower in the later phase of the evolutionary process. Therefore, a dynamic change of interaction frequency is desirable during a parallel CCEA run.

As an answer to question 3, the relationship between interaction frequency and communication cost is non-linear and complicated. The trend is that the best interaction period is prone to larger values when communication cost becomes higher. For problems with a fixed communication cost, the best interaction frequency represents the best trade-off between frequent communication and increased communication overhead, which varies from problem to problem.

The answers to these questions are given based on the 2-D test functions used in our experiments. In future, we will consider high-dimensional test functions. Moreover, the studies were done using a fixed interaction frequency. In reality, a dynamic or adaptive one might be more appropriate, which is another direction of our future work.

Acknowledgments. This work was supported by Honda Research Institute Europe (HRI-EU) and European Commission FP7 grants (Nos. 257906 and 247619).

References

1. Van den Bergh, F., Engelbrecht, A.P.: A cooperative approach to particle swarm optimization. IEEE Transactions on Evolutionary Computation 8(3), 225–239 (2004)
2. Bull, L.: Evolutionary computing in multi-agent environments: Partners. In: Proceedings of the 7th International Conference on Genetic Algorithms. pp. 370–377. Morgan Kaufmann (1997)
3. Jansen, T., Wiegand, R.P.: Sequential versus parallel cooperative coevolutionary (1+ 1) eas. In: Proceedings of the 2003 IEEE Congress on Evolutionary Computation (CEC'03). vol. 1, pp. 30–37. IEEE (2003)

4. Khare, V.R., Yao, X., Sendhoff, B.: Credit assignment among neurons in co-evolving populations. In: Proceedings of the 8th Conference on Parallel Problem Solving from Nature–PPSN VIII. pp. 882–891. Springer (2004)
5. Li, X., Yao, X.: Cooperatively coevolving particle swarms for large scale optimization. IEEE Transactions on Evolutionary Computation 16(2), 210–224 (2012)
6. Mei, Y., Li, X., Yao, X.: Cooperative co-evolution with route distance grouping for large-scale capacitated arc routing problems. IEEE Transactions on Evolutionary Computation 18(3), 435–449 (2014)
7. Panait, L., Luke, S.: Time-dependent collaboration schemes for cooperative coevolutionary algorithms. In: Proceedings of the 2005 AAAI Fall Symposium on Coevolutionary and Coadaptive Systems (2005)
8. Popovici, E., De Jong, K.: A dynamical systems analysis of collaboration methods in cooperative co-evolution. In: AAAI Fall Symposium on Coevolutionary and Coadaptive Systems (2005)
9. Popovici, E., De Jong, K.: The effects of interaction frequency on the optimization performance of cooperative coevolution. In: Proceedings of the 2006 Genetic and Evolutionary Computation Conference (GECCO'06). pp. 353–360. ACM (2006)
10. Potter, M.A.: The design and analysis of a computational model of cooperative coevolution. Ph.D. thesis, George Mason University (1997)
11. Potter, M.A., De Jong, K.A.: A cooperative coevolutionary approach to function optimization. In: Proceedings of the 3rd Conference on Parallel Problem Solving from Nature–PPSN III, pp. 249–257. Springer (1994)
12. Ray, T., Yao, X.: A cooperative coevolutionary algorithm with correlation based adaptive variable partitioning. In: Proceedings of the 2009 IEEE Congress on Evolutionary Computation (CEC'09). pp. 983–989. IEEE (2009)
13. Shi, Y.j., Teng, H.f., Li, Z.q.: Cooperative co-evolutionary differential evolution for function optimization. In: Advances in natural computation, pp. 1080–1088. Springer (2005)
14. Suganthan, P.N., Hansen, N., Liang, J.J., Deb, K., Chen, Y.P., Auger, A., Tiwari, S.: Problem definitions and evaluation criteria for the cec 2005 special session on real-parameter optimization. KanGAL Report 2005005 (2005)
15. Sun, L., Yoshida, S., Cheng, X., Liang, Y.: A cooperative particle swarm optimizer with statistical variable interdependence learning. Information Sciences 186(1), 20–39 (2012)
16. Tan, K.C., Yang, Y., Goh, C.K.: A distributed cooperative coevolutionary algorithm for multiobjective optimization. IEEE Transactions on Evolutionary Computation 10(5), 527–549 (2006)
17. Wiegand, R.P.: An analysis of cooperative coevolutionary algorithms. Ph.D. thesis, George Mason University (2003)
18. Wiegand, R.P., Liles, W.C., De Jong, K.A.: An empirical analysis of collaboration methods in cooperative coevolutionary algorithms. In: Proceedings of the Genetic and Evolutionary Computation Conference (GECCO'01). vol. 2611, pp. 1235–1245 (2001)
19. Wiegand, R.P., Liles, W.C., De Jong, K.A.: Analyzing cooperative coevolution with evolutionary game theory. In: Proceedings of the 2002 IEEE Congress on Evolutionary Computation (CEC'02). vol. 2, pp. 1600–1605. IEEE (2002)
20. Yang, Z., Tang, K., Yao, X.: Differential evolution for high-dimensional function optimization. In: Proceedings of the 2007 IEEE Congress on Evolutionary Computation (CEC'07). pp. 3523–3530. IEEE (2007)
21. Yang, Z., Tang, K., Yao, X.: Large scale evolutionary optimization using cooperative coevolution. Information Sciences 178(15), 2985–2999 (2008)
22. Yang, Z., Zhang, J., Tang, K., Yao, X., Sanderson, A.C.: An adaptive coevolutionary differential evolution algorithm for large-scale optimization. In: Proceedings of the 2009 IEEE Congress on Evolutionary Computation (CEC'09). pp. 102–109. IEEE (2009)