Minimizing Class-Based Completion Time Variance on a Single Machine

Xueping Li
Yuerong Chen
University of Tennessee

Yang Sun
Arizona State University

Outline

- Background & motivation
- Introduction to CB-CTV minimization problems
- Properties & algorithms
- Computational results
- Concluding remarks

Background

- Limited study on non-regular performance optimization
  - **Completion Time Variance (CTV)**
- Stabilized service
- JIT philosophy
  - Penalize both ***E*** & ***T***.
- Applications
  - Internet data packet dispatching
  - Jitter (video/audio communication)

Introduction to 1||CTV

- The single machine CTV minimization problems were first proposed for the computer file organization problem (Merten and Muller, 1972)
- The goal is to provide jobs with a uniform or fair treatment
- Computation of the overall CTV:

$$CTV = \frac{1}{n-1}\sum_{i=1}^{n}(C_i - \overline{C})^2$$

where n is the number of jobs, C_i is the completion time of the i^{th} job, and $\overline{C} = \frac{1}{n}\sum_{i=1}^{n} C_i$.

- Example:

Job #:	1	2	3	4	5	6	7	8	9
Processing Time:	12,	1,	4,	20,	14,	8,	5,	16,	2
Completion Time:	12,	13,	17,	37,	51,	59,	64,	80,	82
CTV:	767.1111								

Lit. Review

- **1 || CTV** is NP-hard (Kubiak, 1993)
- Waiting Time Variance (WTV) is antithetical to Completion Time Variance (CTV) (Mertern & Muller, 1972)
- The longest jobs must be the first one to process (Schrage, 1975)
- There exists one optimal sequence like ***1**st, 2nd….,3rd* (Hall & Kubiak, 1991)
- Optimal sequences are ***V-Shaped*** which means that the jobs before the shortest job are descending sorted while the jobs after are ascending sorted (Vani & Raghavachari, 1977)

A different point of view

- The **1 || CTV** minimization problems are considered from the viewpoint of the system.
  - In this point of view, jobs are assumed to be independent of each other, which is often not practical in the real world.
  - The system CTV performance measure may result in the dissatisfaction of a certain user with the service
- It is necessary, therefore, to investigate CTV minimization problems from the viewpoint of users. The CB-CTV minimization problem arises accordingly.
  - **1 || CB-CTV**

Introduction to 1 || CB-CTV

- Definition of CB-CTV:

$$CB-CTV = \sum_{i=1}^{L} \frac{n_i}{n} CTV_i$$

where ***L*** is the number of classes, ***n*** is the total number of all jobs, ***n_i*** is the number of jobs in the ***i^{th}*** class, and ***CTV_i*** is the CTV of jobs in the ***i^{th}*** class. ***CB-CTV*** is computed as follows:

$$CTV_i = \frac{1}{n_i - 1} \sum_{j=1}^{n_i} (C_{ij} - \bar{C}_i)^2$$

where ***C_{ij}*** is the completion time of the ***j^{th}*** job in the ***i^{th}*** class and $\bar{C}_i$ is the mean completion time of the jobs in the ***i^{th}*** class.

An illustration of 1||CB-CTV

- One Example:

Class I: 20, 5 ; Class II: 14, 2, 12 ; Class III: 8, 4, 1, 16

A possible schedule: 12, 1, 4, 20, 14, 8, 5, 16, 2

	Class I	Class II	Class III
Completion Time:	37, 64;	51, 82, 12;	59, 17, 13, 80
CTV:	364.5;	1230.3;	1066.3

CB-CTV: 2/9*364.5 + 3/9*1230.3 + 4/9*1066.3 = 965

Proof of Lemmas

- Basic notations:
- An illustration:
  - q^{th} class

p_{ij} = the processing time of the j^{th} processed job in the i^{th} class;

X_i = the i^{th} job block that separates the jobs in a certain class.

$$p_{21}, p_{31}, p_{q1}, p_{q2}, p_{22}, p_{11}, p_{32}, p_{q3}, p_{L1}, p_{23}, p_{q4}, p_{q5}, p_{q6}, \ldots, p_{Ln_L}, p_{qn_q}, p_{36}, p_{2n_2}, \ldots$$

$$\boxed{X_0}, p_{q1}, p_{q2}, \boxed{X_1}, p_{q3}, \boxed{X_2}, p_{q4}, p_{q5}, p_{q6}, \boxed{X_3}, \ldots, \boxed{X_{s-1}}, p_{qn_q}, \boxed{X_s}$$

Lemma 1 *CTV_q is smaller in the following schedule*

$$\boxed{X_0}, p_{q1}, p_{q2}, \ldots, p_{qn_q}, \boxed{X_1}, \boxed{X_2}, \boxed{X_3}, \ldots, \boxed{X_{s-1}}, \boxed{X_s}$$

Lemma 2 *CTV_q keeps a constant, as long as the scheduling form satisfies: i)No jobs from other classes are scheduled among the q^{th} class, i.e., no blocks exist among the q^{th} class; and ii)The inner-class scheduling order of the q^{th} class keeps unchanged.*

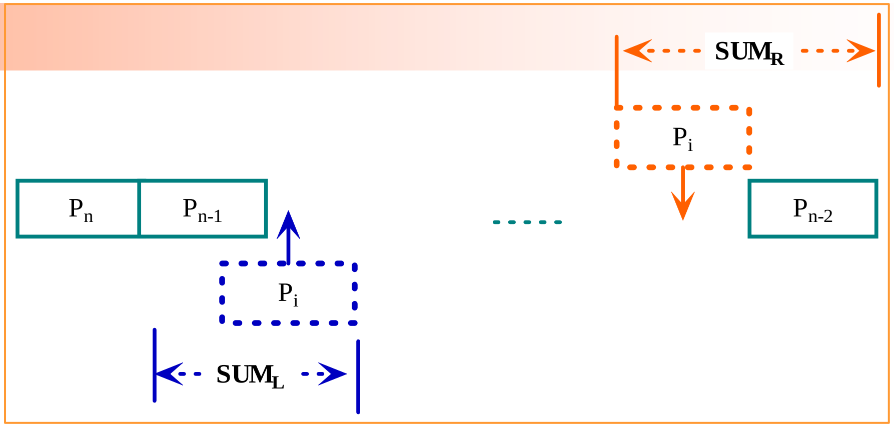
The relationship between CB-CTV & CTV

- *Theorem*: A CB-CTV minimization problem can be transformed into a series of CTV minimization problems. That is, the following equation holds:

$$\underset{\lambda \in \Lambda}{Min}\left(\sum_{i=1}^{L} \frac{n_i}{n} CTV_i(\lambda) \right) = \sum_{i=1}^{L} \left(\frac{n_i}{n} \underset{\lambda_i \in \Lambda_i}{Min}\left(CTV_i(\lambda_i)\right) \right)$$

where λ , Λ, and $CTV_i(\lambda)(i=1,\cdots,L)$ are respectively a schedule of all jobs of L classes, the schedule set composed of all possible λ , and the CTV of the i^{th} class under the schedule λ , while $\lambda_i(i=1,\cdots,L)$, $\Lambda_i(i=1,\cdots,L)$, and $CTV_i(\lambda_i)(i=1,\cdots,L)$ are respectively a schedule of all jobs of the i^{th} class, the schedule set composed of all possible λ_i , and the CTV of the i^{th} class under the schedule λ_i.

Balanced Spiral (BS) Algorithm*



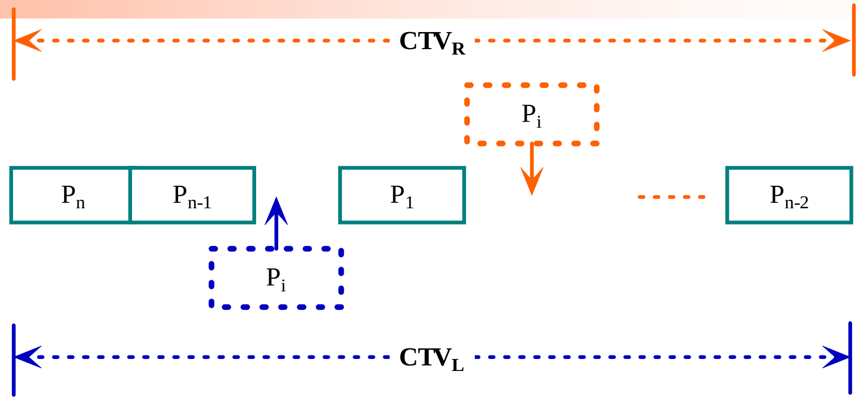
1. Place the job p_n in the first position, the job p_{n-1} in the second position, and the job p_{n-2} in the last position. Let sequence $Lt = \{p_{n-1}\}$ and sequence $Rt = \{p_{n-2}\}$. Denote by SUM_{Lt} and SUM_{Rt} respectively the sums of the processing times of the jobs in Lt and Rt.

2. If $SUM_{Lt} < SUM_{Rt}$, append the largest job from the unscheduled jobs to sequence Lt, and update SUM_{Lt}; If $SUM_{Lt} \geq SUM_{Rt}$, prepend the largest job from the unscheduled jobs to sequence Rt, and update SUM_{Rt}.

3. Repeat Step 2 until all the jobs are scheduled.

•N. Ye, X. Li and T. Farley (2006) "Job Scheduling Methods for Reducing the Variance of Job Waiting Times on Computer Networks", to appear in *Computer and Operations Research*.

Verified Spiral (VS) Algorithm*



1. According to Schrage's conjecture, place the job p_n in the first position, the job p_{n-1} in the second position, and the job p_{n-2} in the last position. The shortest job p_1 is placed in the position between p_{n-1} and p_{n-2}.

2. Select the longest job from the unscheduled jobs. Place it either exactly before the job p_1 or exactly after the job p_1, depending on which way produces a smaller CTV of the job sequence so far.

3. Repeat Step 2 until all the jobs are scheduled.

•N. Ye, X. Li and T. Farley (2006) "Job Scheduling Methods for Reducing the Variance of Job Waiting Times on Computer Networks", to appear in *Computer and Operations Research*.

Computational results (small-size)

Table 3: CB-CTV vs. CTV for four small problem instances. It shows consistently smaller CTV for individual class under CB-CTV minimization than under the overall CTV minimization.

No.	Optimal sequences	CTV_1	CTV_2	CTV_3	CTV_4	CTV_5	$CB\text{-}CTV$	CTV
1	CB: 19, 12, 7, 11, 13, 3, 15, 6, 5	4.5*	30.33*	158.25*	N/A	N/A	81.44*	644.11
	NCB: 19, 15, 11, 7, 3, 5, 6, 12, 13	648	289.33	588.33	N/A	N/A	501.93	476.78*
2	CB: 20, 4, 8, 16, 13, 1, 6, 18, 15	112.5*	37.33*	70.92*	N/A	N/A	68.96*	761.19
	NCB: 20, 16, 15, 6, 4, 1, 8, 13, 18	1250	710.33	372.33	N/A	N/A	680.04	572.61*
3	CB: 20, 2, 16, 5, 10, 6, 11, 7, 12, 9	18*	2*	40.5*	24.5*	12.5*	19.5*	716.1
	NCB: 20, 12, 11, 9, 5, 2, 6, 7, 10, 16	144.5	760.5	200	420.5	840.5	473.2	533.78*
4	CB: 19, 9, 20, 5, 17, 1, 13, 10, 18, 16	12.5*	50*	128*	0.5*	40.5*	46.3*	1226.01
	NCB: 20, 18, 17, 10, 9, 1, 5, 13, 16, 19	1800	392	2520.5	200	1458	1274.1	1021.34*

Computational results (large-size)

Table 5: Performance comparison of CB-CTV and overall CTV for eight large problem instances.

No.	CTV^1_{Im}	CTV^2_{Im}	CTV^3_{Im}	CTV^4_{Im}	CTV^5_{Im}	$CB\text{-}CTV_{Im}$	CTV_S
1 (VS)	95.31%	92.39%	73.3%	N/A	N/A	81.7%	47.35%
2 (VS)	96.07%	89.1%	75.66%	N/A	N/A	85.05%	46.03%
3 (BS)	95.12%	90.64%	77.12%	N/A	N/A	85.2%	50.41%
4 (BS)	95.85%	90.64%	74.83%	N/A	N/A	85.61%	43.54%
5 (VS)	95.18%	96.67%	96.23%	94.09%	95.97%	95.64%	48.72%
6 (VS)	96.07%	95.71%	95.61%	95.63%	95.06%	95.69%	51.8%
7 (BS)	94.8%	96.58%	94.91%	95.38%	96.15%	95.53%	45.7%
8 (BS)	95.63%	94.72%	97.04%	95.51%	94.24%	95.51%	48.45%

$$CTV^i_{Im} = \frac{CTV^i_{NCB} - CTV^i_{CB}}{CTV^i_{NCB}} * 100\% \qquad i = 1, 2, \ldots, 5$$

$$CB\text{-}CTV_{Im} = \frac{CB\text{-}CTV_{NCB} - CB\text{-}CTV_{CB}}{CB\text{-}CTV_{NCB}} * 100\%$$

$$CTV_S = \frac{CTV_{CB} - CTV_{NCB}}{CTV_{NCB}} * 100\%$$

In conclusion …

- CB-CTV is closely related to service stability since it penalizes both earliness and tardiness, and it is further related to customer satisfaction because it takes into account customer preferences.
- CB-CTV minimization has wide applications in many areas such as packet scheduling for Internet communications and reservation systems, modern manufacturing systems, supply chain management, and others where it is desirable to achieve service stability while considering customer preference.

Conclusion

- A CB-CTV minimization problem can be transformed into a series of CTV minimization problems. This transformation dramatically simplifies the problem since there have been a lot of heuristics that can be used for CTV minimization problems.

- There is a trade-off between the overall CTV and CB-CTV. However, the rate of the overall CTV performance sacrifice is much smaller than that of CB-CTV performance improvement. It indicates that CB-CTV is desirable.

X. Li, Y. Chen, Y. Sun and R. Sawhney (2006) "On the Minimization of Class-based Completion Time Variance", to appear in *International Journal of Operations Research*.

Q & A

Thanks!