Homework Set 2

Exercise 9.2

Page 132

Online Computation And Competitive Analysis

Borodin & El-Yaniv

Statement of Exercise

Given an N-state MTS, an initial state, and a task sequence of length *n*, what are the *time and space* complexities to compute the optimal off-line *schedule and cost* to serve the task sequence using the [above] work function method?

Short Answer

The time and space complexities of generation and traceback of the dynamic programming table for offline-WFA.

Task Type Processing Cost

In general:

- Single task of given ***type*** could incur a different processing cost in each of ***N*** MTS states.
- **T** = {task types}: |**T**| = $T \leq N$.
- *Represent as a function:*

  $$ttpc : (\mathbf{T} \times \mathbf{N}) \rightarrow \mathbb{R}^+ \ (0 \notin \mathbb{R}^+)$$

- $|ttpc| = |\mathbf{T}| \cdot |\mathbf{V}| = T \cdot N \leq N \cdot N = N^2$

Task Type Processing Cost

"TTPC" Table: *Storage* = |**T**|·|**V**| = ***T***·***N*** ≤ ***N***^{2}

Task-type Processing Cost *(for Task Type By MTS State)*

Task Type	*MTS State* 1	2	3	● ● ●	*s*	● ● ●	*N*
1	*r*(1,1)	*r*(1,2)	*r*(1,3)	●	*r*(1,*s*)	●	*r*(1,*N*)
2	*r*(2,1)	*r*(2,2)	*r*(2,3)	●	*r*(2,*s*)	●	*r*(2,*N*)
●	●	●	●	●	●	●	●
t	*r*(*t*,1)	*r*(*t*,2)	*r*(*t*,3)	●	*r*(*t*,*s*)	●	*r*(*t*,*N*)
●	●	●	●	●	●	●	●
T	*r*(*T*,1)	*r*(*T*,2)	*r*(*T*,3)	●	*r*(*T*,*s*)	●	*r*(*T*,*N*)

Inter-state Transition Cost

Define "***measurable (point-) space***":

["**space** (of vertices/points) is ***measurable***"] → [∃ a "*measurement function **d***" that assigns a numeric value (signed magnitude of uniform units) to every vertex pair.

Inter-state Transition Cost

In a general "*measurable space*":

- Measure function is *not* necessarily *symmetric*:

$$(\exists x)(\exists y)[\boldsymbol{d}(x,y) \neq \boldsymbol{d}(y,x)],\ x,y \in \mathbf{V}$$

- Such a "*measurable space*" is characterized as a "***non metric-***space", and
- $|\boldsymbol{d}| = |\mathbf{V}| \cdot |\mathbf{V}| = \boldsymbol{N} \cdot \boldsymbol{N}$

ITC: Non Metric-Space

"ITC" ($\boldsymbol{d}$) Table: *Storage* = $|\mathbf{V}| \cdot |\mathbf{V}| = \boldsymbol{N} \cdot \boldsymbol{N} = \boldsymbol{N}^2$

Inter-state Transition Cost (*MTS Space NOT a Metric Space*)

MTS State	*MTS State* 1	2	3	•••	*s*	•••	*N*
1	*d*(1,1)	*d*(1,2)	*d*(1,3)	•	*d*(1,*s*)	•	*d*(1,*N*)
2	*d*(2,1)	*d*(2,2)	*d*(2,3)	•	*d*(2,*s*)	•	*d*(2,*N*)
•	•	•	•	•	•	•	•
s	*d*(*s*,1)	*d*(*s*,2)	*d*(*s*,3)	•	*d*(*s*,*s*)	•	*d*(*s*,*N*)
•	•	•	•	•	•	•	•
N	*d*(*N*,1)	*d*(*N*,2)	*d*(*N*,3)	•	*d*(*N*,*s*)	•	*d*(*N*,*N*)

Inter-state Transition Cost

But if the measure function **IS** *symmetric* -

$$(\forall x)(\forall y)[\boldsymbol{d}(x,y) = \boldsymbol{d}(y,x)],\ x,y \in \mathbf{V}:$$

Such a "*measurable space*" is characterized as a "***metric space***", and

$$|d| = \frac{N(N+1)}{2} = \left(\frac{N}{2}+1\right)N$$

ITC: Metric-Space

"ITC" (***d***) Table: *Storage* $= \frac{N(N+1)}{2}$

Inter-state Transition Cost (*MTS Space **IS a Metric Space***)							
MTS State	*MTS State*						
	1	**2**	**3**	● ● ●	***s***	● ● ●	*N*
1	***d***(1,1)	***d***(1,2)	***d***(1,3)	●	***d***(1,*s*)	●	***d***(1,*N*)
2		***d***(2,2)	***d***(2,3)	●	***d***(2,*s*)	●	***d***(2,*N*)
●			●	●	●	●	●
s					***d***(*s*,*s*)	●	***d***(*s*,*N*)
●						●	●
N							***d***(*N*,*N*)

ITC: Metric-Space

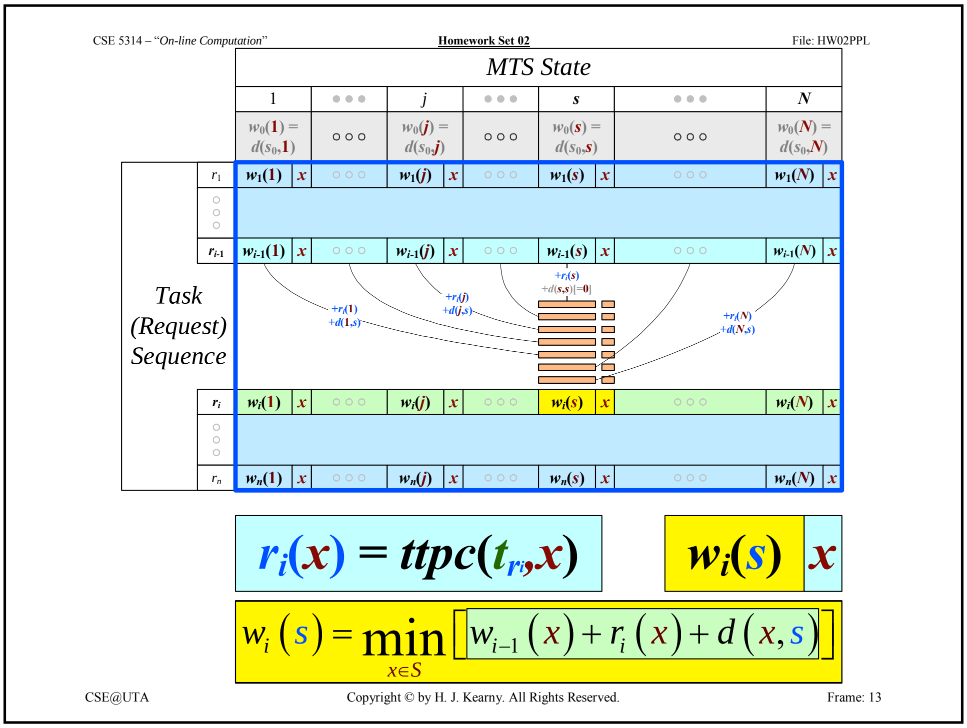
Important comments:

- Foregoing explicitly includes:

$$(\forall x)[\mathbf{d}(x,x)],\ x \in \mathbf{V}$$

- No explicit claim made concerning value of any such member.
- Guarantee that Triangle Inequality does not fail is enforced if $(\forall x)[\mathbf{d}(x,x) = 0],\ x \in \mathbf{V}$ **and at risk otherwise.**
- Algorithmic-time-wise, "cost" of lookup returning 0 traded for cost of compare ($x = x$) computing 0.

Offline-WFA-computation DP Table



CSE 5314 – "*On-line Computation*" **Homework Set 02** File: HW02PPL

Offline-WFA-computation DP Table

Storage Requirements:

- Length(σ) = n
- |**V**| = N
- "$w_i(s)|_x$" & x: **2** storage items

Total storage for this table:

$2nN$

CSE@UTA Copyright © by H. J. Kearny. All Rights Reserved. Frame: 14

Toward Total Storage Requirement

- Task Type Processing Cost Table:

$$|ttpc| = T \cdot N \; (\leq N^2)$$

- Inter-state Transition Cost Table:

$$|d|_{not\ metric} = N \cdot N$$

$$|d_{metric}| = \frac{N(N+1)}{2} = \left(\frac{N}{2}+1\right)N$$

- Offline-WFA-Computation DP Table:

$$2nN$$

*Storage Requirement: **Not-metric** Space*

Total storage = |***ttpc***|+|***d***|+|***DP Table***|:

$$Actual = TN + NN + 2nN$$

$$= 1 \cdot N^2 + (2n+T)N$$

$$Limit_{T=N} = NN + NN + 2nN$$

$$= 2 \cdot N^2 + (2n)N$$

$$= Actual + (N-T)N$$

*Storage Requirement: **Metric** Space*

Total storage = |***ttpc***|+|***d***|+|***DP Table***|:

$$Actual = TN + N\left(\frac{N}{2}+1\right) + 2nN = \frac{N^2}{2} + TN + N + 2nN$$

$$= \left(\frac{1}{2}\right)N^2 + \left(2n+1+T\right)N$$

$$Limit = \frac{N^2}{2} + N^2 + N + 2nN = \left(\frac{1}{2}+1\right)N^2 + \left(2n+1\right)N$$

$$= Actual + \left(N-T\right)N$$

Storage Requirement Recap

Storage requirements then are of the form

$$aN^2+bN$$

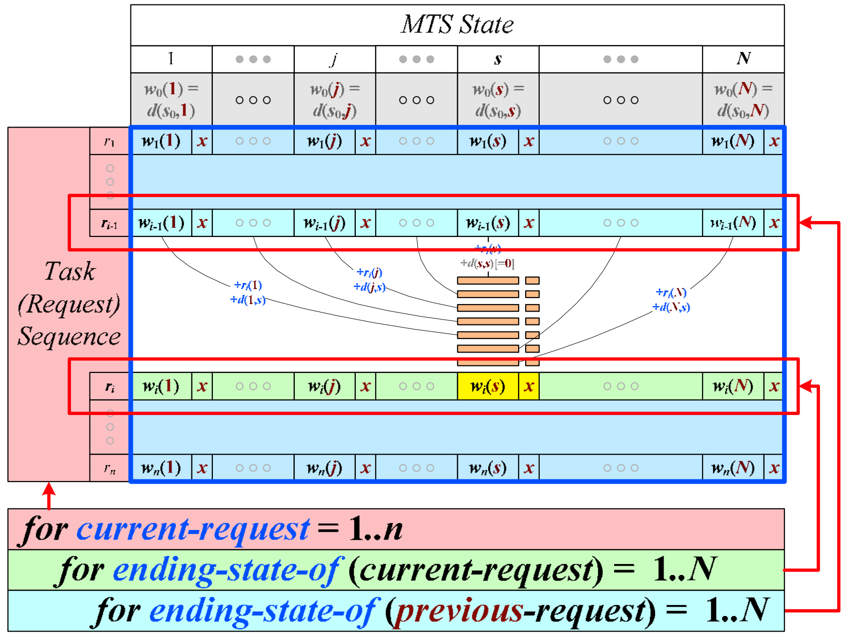
	NOT Metric-Space	***IS Metric-Space***
Actual	$1 \cdot N^2 + \left(2n+T\right)N$	$\frac{1}{2}N^2 + \left(2n+T+1\right)N$
$\text{Limit}_{T=N}$	**Actual + (*N*-*T*)*N***	**Actual + (*N*-*T*)*N***

*Computation: Load **TTPC** & **ITC***

- Both items part of problem specification.
- Computation per se composed solely of "loading", prior to subsequent look-up use.
- Thus, expression for time complexity equivalent to that for storage requirement:
- TTPC: $T \cdot N$ ($\leq N^2$)
- ITC: *Not metric* – $N \cdot N$

  Metric – $\left(\frac{N}{2}+1\right)N$

Computation: Offline-WFA-DP Table



Computation: Offline-WFA-DP Table

- All three loops must be completely passed (no shortcuts).
- Computation time:

$$nNN = nN^2$$

Computation: Optimal Cost & Schedule

- Optimal cost of serving entire sequence is value of minimal entry in final row of DP table.
- Computation time is time to pass final row:

$$N$$

Offline-computation DP Table for MTS

Task Sequence σ and Prefixes σ_i

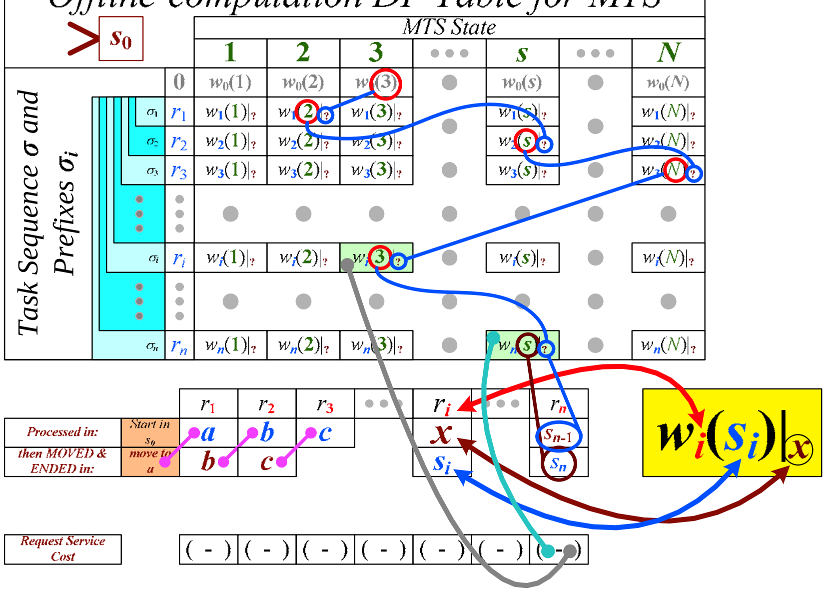
$> s_0$		MTS State						
		1	**2**	**3**	• • •	**s**	• • •	**N**
	0	$w_0(1)$	$w_0(2)$	$w_0(3)$	•	$w_0(s)$	•	$w_0(N)$
σ_1	r_1	$w_1(1)\vert_?$	$w_1(2)\vert_?$	$w_1(3)\vert_?$	•	$w_1(s)\vert_?$	•	$w_1(N)\vert_?$
σ_2	r_2	$w_2(1)\vert_?$	$w_2(2)\vert_?$	$w_2(3)\vert_?$		$w_2(s)\vert_?$		$w_2(N)\vert_?$
σ_3	r_3	$w_3(1)\vert_?$	$w_3(2)\vert_?$	$w_3(3)\vert_?$	•	$w_3(s)\vert_?$	•	$w_3(N)\vert_?$
⋮	⋮	•	•	•	•	•	•	•
σ_i	r_i	$w_i(1)\vert_?$	$w_i(2)\vert_?$	$w_i(3)\vert_?$	•	$w_i(s)\vert_?$	•	$w_i(N)\vert_?$
⋮	⋮	•	•	•	•	•	•	•
σ_n	r_n	$w_n(1)\vert_?$	$w_n(2)\vert_?$	$w_n(3)\vert_?$	•	$w_n(s)\vert_?$	•	$w_n(N)\vert_?$

Computation: Optimal Cost & Schedule

- Schedule of service and other desired intermediate results extracted from recursive traceback through DP table.
- Ancillary activities (e.g., constructing a secondary “results” structure for some need) are ignored.
- Computation time is time to pass back to a single entry in all rows:

$$n$$

Offline-computation DP Table for MTS



Toward Total Computation Time

- Startup loading - TTPC: $T \cdot N$ ($\le N^2$)
- Startup loading - ITC:

  Not metric – $N \cdot N$

  Metric – $\left(\frac{N}{2}+1\right)N$
- Offline-WFA-DP Computation: nN^2
- Find optimal cost: N
- Trace-back for service schedule: n

Toward Total Computation Time

For a space that is NOT metric:

$$\begin{aligned}
Actual &= TN + N^2 + nN^2 + N + n \\
&= (n+1)N^2 + (1+T)N + n \\
Limit_{T=N} &= N^2 + N^2 + nN^2 + N + n \\
&= (n+1+1)N^2 + (1)N + n \\
&= Actual + (N-T)N
\end{aligned}$$

Toward Total Computation Time

For a metric space:

$$Actual = TN + \left(\frac{N}{2}+1\right)N + nN^2 + N + n$$

$$= TN + \frac{N^2}{2} + N + nN^2 + N + n$$

$$= \left(n+\frac{1}{2}\right)N^2 + (T+2)N + n$$

$$Limit_{T=N} = N^2 + \frac{N^2}{2} + N + nN^2 + N + n$$

$$= \left(n+\frac{1}{2}+1\right)N^2 + (2)N + n$$

$$= Actual + (N-T)N$$

Storage & Time Recap

- In both storage & time:

$$\boldsymbol{Limit_{T=N} = \text{Actual} + (N-T)N}$$

	NOT Metric-Space	***IS Metric-Space***
Actual Storage	$1 \cdot N^2 + (2n+T)N$	$\frac{1}{2}N^2 + (2n+T+1)N$
Actual Time	$(n+1)N^2 + (1+T)N + n$	$\left(n+\frac{1}{2}\right)N^2 + (T+2)N + n$

*Perspective: Who dominates **n** or **N**?*

In a ***k***-server example with ***k*** servers in a space of ***V*** vertices:

- May expect number of MTS states *N* to approximate number of combinations "***V*** choose ***k***" (if bar more than one server at a point):

$$N = \left(\frac{V!}{k!(V-k)!} \right)$$

*Perspective: Who dominates **n** or **N**?*

- For metric space storage:

$$\begin{aligned} Storage &= \frac{1}{2}N^2 + (2n + T + 1)N \\ &= FixedOverhead + DyamicCost \\ &= \left[\frac{1}{2}N^2 + (T+1)N\right] + \left[(2N)n\right] \end{aligned}$$

*Perspective: Who dominates **n** or **N**?*

Assume modestly sized example:

- $V = 20$ points in a metric space
- $k = 5$ servers
- $T = 7$ types of request (different proc. costs)

For N MTS states:

$$N = \left(\frac{V!}{k!(V-k)!} \right) = \frac{20!}{5!(20-5)!} = \frac{20!}{5!(15)!}$$

$$= \frac{16 \cdot 17 \cdot 18 \cdot 19 \cdot 20}{2 \cdot 3 \cdot 4 \cdot 5} = 16 \cdot 17 \cdot 3 \cdot 19 = 15{,}504$$

*Perspective: Who dominates **n** or **N**?*

How many storage locations are required:

$$Storage = FixedOverhead + DyamicCost$$

$$= \left[\frac{1}{2} N^2 + (T+1) N \right] + \left[(2N) n \right]$$

$$= \left[\frac{1}{2} (15504)^2 + (7+1)(15504) \right] + \left[(2 \cdot 15504) n \right]$$

$$= (15504)[7760] + \left[(31008) n \right]$$

$$= 120{,}311{,}040 + (31008) n$$

*Perspective: Who dominates **n** or **N**?*

Space if *sizeof*(storage-location) = 4 bytes:

$$
\begin{aligned}
StorageSpace &= FixedOverheadSpace + DyamicCostSpace \\
&= 4 \cdot \left[120{,}311{,}040 + (31{,}008)n\right] \\
&= 481{,}244{,}160 + (124{,}032)n \\
1Gb &= 1{,}073{,}741{,}824 \\
1{,}073{,}741{,}824 &= 481{,}244{,}160 + 124{,}032n \\
n &= \frac{1073741824 - 481244160}{124032} \\
&= \frac{592{,}497{,}664}{124032} \sim 4{,}777 requests
\end{aligned}
$$

*Perspective: Who dominates **n** or **N**?*

Assume our idiot box has 1 gig of memory. We should be able to whip out 4,777 requests pretty quickly….

Perspective: Who dominates *n* or *N*?

Can determine number of time (calculation) units using number of storage units:

$$Time = \left(\frac{1}{2} + n\right)N^2 + (T+2)N + n$$

$$Storage = \left(\frac{1}{2}\right)N^2 + (2n+T+1)N$$

$$Time = Storage + ConversionFactor$$

$$= Storage + \left(nN^2 - 2nN + N + n\right)$$

Perspective: Who dominates *n* or *N*?

$$Time = Storage + \left(nN^2 - 2nN + N + n\right)$$

$$= Storage + \left(nN^2 - (2n-1)N + n\right)$$

$$= \left[120,311,040 + (31,008)n\right] + \left(n(15504)^2 - (2n-1)(15504) + n\right)$$

$$= 120,311,040 + (31,008)n + n(15504)^2 - (2n-1)(15504) + n$$

$$= 120,311,040 + (31,008)n + n(15504)^2 - 31008n + (15504) + n$$

$$= 120,311,040 + n(15504)^2 + (15504) + n$$

$$= 120,311,040 + n(240374016) + (15504) + n$$

$$= 120,326,544 + n(240,374,017)$$

$$= 120,326,544 + (4,777)(240,374,017)$$

$$= 120,326,544 + (4,777)(240,374,017)$$

$$= 1,148,387,005,753 \text{ calculation units}$$

*Perspective: Who dominates **n** or **N**?*

If 1,000,000 calculations occur per second, this problem will only take:

13 seconds less than 13days + 7 hours

Assuming, after squinting at the dim console lights for a couple hours, you don't then reboot figuring the machine's hung…

(Parallel/distributed, anyone?)

End of Exercise 9.2 Page 132

