Market Rule Amendment Proposal Form

Part 1 - Market Rule Information

Identification No.:	MR-00494-R00
Subject:	Pseudo-Unit Updates in the Calculation Engines
Title:	Pseudo-Unit Updates in the Calculation Engines
Nature of Proposal:	☒ Alteration ☐ Deletion ☒ Addition
Chapter:	7
Appendix:	7.5 – The Day-Ahead Market Calculation Engine Process 7.5A - The Pre-Dispatch Calculation Engine Process 7.6 - The Real-Time Calculation Engine Process
Sections:	Various
Sub-sections proposed for amending:	Various
Current Market Rules Baseline:	55.1

Part 2 - Proposal History

Version	Reason for Issuing	Version Date
1.0	Draft for Stakeholder review	July 7, 2026
2.0	Draft for Technical Panel review	October 7, 2026
3.0	Draft for Stakeholder review and comment	October 15, 2026
4.0	Submitted for Technical Panel vote	November 3, 2026

Approved Amendment Publication Date:

Approved Amendment Effective Date:

Part 3 - Explanation for Proposed Amendment

Provide a brief description that includes some or all of the following points:

- The reason for the proposed amendment and the impact on the *IESO-administered markets* if the amendment is not made.
- Alternative solutions considered.
- The proposed amendment, how the amendment addresses the above reason and impact of the proposed amendment on the *IESO-administered markets*.

Summary

The IESO proposes to amend the market rules to improve how pseudo-units (PSUs) are scheduled in day-ahead, pre-dispatch and real-time, by more accurately reflecting outages, de-rates, constraints, and ramp rates within the Dispatch Scheduling Optimizer (DSO) when producing dispatch outcomes.

Since the launch of the renewed market, situations have emerged where the scheduling logic for these resources produces infeasible dispatch outcomes. Three enhancements have been identified to improve the scheduling logic:

1- Improve the treatment steam turbine de-rate allocations when there are outages on the combustion turbine(s);
2- Recognize de-rates and constraints below calculated PSU minimum loading points; and
3- Improve the ramp rate modelling when combustion turbines and steam turbines synchronize at different times.

The amendments are intended to enhance how these resources are scheduled and resolve situations where current scheduling logic can produce infeasible dispatch outcomes.

Further information can be found on the Pseudo-Unit Updates stakeholder engagement webpage.

Background

Following the launch of the renewed market, the IESO identified several discrepancies in the scheduling of combined-cycle generating facilities modelled as PSUs. Under certain operating conditions, the existing DSO logic could produce infeasible schedules that do not accurately reflect the physical capabilities or operating state of the generating units.

The first issue relates to how steam turbine (ST) de-rates are allocated across PSUs. Currently, the DSO allocates ST de-rates across all associated PSUs who have CT offers present, including PSUs that may be unavailable due to submitted combustion turbine (CT) outages. This can reduce the ST's actual available capability, and in some circumstances, cause a PSU to be dispatched to zero output.

The second issue arises when CT or ST de-rates and operating constraints reduce the available capability below the DSO-calculated PSU minimum loading point (MLP). Under the current logic, the PSU may be deemed unavailable and dispatched to zero output, even when operation below the calculated MLP remains physically achievable. This issue is not limited to PSUs and can also affect non-quick-start (NQS) resources.

The third issue concerns ramp rate calculations when combustion turbines and steam turbines synchronize at different times (i.e. ST syncs after CT). In these situations, the real-time DSO calculates ramp trajectories for the ST using the ST's output level as the initialization level without considering the CT's output. This can cause the overall PSU to ramp more quickly than intended, resulting in unrealistic scheduling outcomes.

Discussion

The IESO has proposed three enhancements to address the identified issues and improve alignment between dispatch schedules and the physical operation of combined-cycle facilities.

Enhancement 1: CT Outages and Allocation of ST De-rates

The first enhancement introduces logic that considers CT outage information when allocating ST de-rates and maximum constraints across associated PSUs. The objective is to ensure that de-rates are only applied to eligible PSUs and are not allocated to units that are already unavailable due to outages. The amendment applies across the Day-Ahead Market, Pre-Dispatch, and Real-Time scheduling timeframes. Several new and modified market rule provisions are proposed to formalize the eligibility criteria and revised prorating methodology.

Added sections:

- 22.2.2A (Appendix 7.5),
- 15.2.2A (Appendix 7.5A),
- 10.2.2A (Appendix 7.6),
- 22.3.5.2A (Appendix 7.5),
- 15.3.5.2A (Appendix 7.5A),
- 10.3.5.1A (Appendix 7.6)

Description: These new sections have been incorporated to clarify the treatment of de-rates and constraints of STs in the Day-Ahead Market (DAM), Pre-dispatch (PD), and Real-Time (RT). Specifically, they establish the eligibility criteria that a PSU must meet, so that the prorating of the most limiting ST maximum constraint or ST de-rate across associated PSUs and CTs is applied.

Modified sections:
- 22.2.3.1.2 (Appendix 7.5),
- 22.3.5.3 (Appendix 7.5),
- 15.2.3.1.2 (Appendix 7.5A),
- 15.3.5.3 (Appendix 7.5A),
- 10.2.3.1.2 (Appendix 7.6),
- 10.2.3.1.6 (Appendix 7.6),
- 10.3.5.2 (Appendix 7.6)

Description: These sections have been adjusted so that the prorating logic only applies when conditions set out in the newly added sections are met.

Modified section:
- 10.2.3.1.5 (Appendix 7.6)

Description: This section was modified to apply CT derates for the associated PSU, when the derate or constraint is below the MLP and the PSU is not operating in single cycle mode.

Added section:
- 10.2.3.1.5A (Appendix 7.6)

Description:
The added section was introduced to capture the scenario where the PSU is operating in single cycle mode or the associated ST is offline.

Modified sections:
- 22.2.3.1.1 (Appendix 7.5),
- 15.2.3.1.1 (Appendix 7.5A),
- 10.2.3.1.1 (Appendix 7.6)

Description: These sections have been modified to incorporate logic for calculating the amount of offered energy attributed to the ST portion. Specially, when a PSU is flagged to operate in single cycle mode, it will now affect the offered energy attributed to the steam turbine portion.
Modified section:

- 10.2.3.1.3 (Appendix 7.6)

Description: This section was modified such that a PSU is no longer deemed unavailable when either: The CT capacity is below its MLP; or the ST capacity associated with the PSU is below its MLP.

Enhancement 2: De-rates and Constraints Below Minimum Loading Point (MLP)

The second enhancement addresses situations where de-rates or operating constraints reduce available capability below the DSO-calculated MLP. Under the proposed approach, the real-time calculation engine will continue to respect submitted de-rates and constraints rather than automatically dispatching the resource to zero output. New market rule provisions establish the logic for determining adjusted CT and ST maximum capabilities and outline how those limits will be applied under different operating configurations, including when units are offline but have confirmed future start-up times. This change applies to real-time scheduling and also affects NQS resources.

Added section:
- 10.3.3.2 (Appendix 7.6)

Description: This section was added to define the logic when a PSU is not operating in single cycle mode and either the maximum constraint or maximum offered quantity is below the MLP. In these circumstances, the real-time calculation engine will adjust the maximum constraints of the associated CT (CTMax) and ST (PRSTMax) accordingly.

Added section:
- 10.3.3.2.1 (Appendix 7.6)

Description: Building upon Section 10.3.3.2, this section was added to define the conditions and corresponding logic when the associated CT is online while the ST is offline.

Added section:
- 10.3.3.2.2 (Appendix 7.6)

Description: Building upon Section 10.3.3.2, this section was added to define the conditions and corresponding logic when both the associated CT and ST are online.

Added section:
- 10.3.3.2.3 (Appendix 7.6)

Description: Building upon Section 10.3.3.2, this section was added to define the conditions under which the associated CT and/or ST is not yet online but have a confirmed future start-up time. Specifically, this section outlines how the calculated maximums shall be applied in the advisory intervals.

Enhancement 3: Ramp Rates for Staggered CT and ST Start-Ups

The third enhancement improves how ramp trajectories are calculated when CTs and STs come online at different times. The proposal shifts the ramp rate calculation to a PSU-based approach, ensuring that the output already being produced by the CT is reflected when the ST subsequently synchronizes. Similarly, when additional CTs synchronize after the ST is already online, the applicable PSU ramp rates will be used to better reflect actual facility capabilities.

Modified section:
- 10.3.4.1 (Appendix 7.6)

Description: This section was modified to include the condition that the steam turbine resource is online.

Modified section:
- 10.3.4.2 (Appendix 7.6)

Description: This section was modified to include the condition that the steam turbine resource is offline.

Modified section:
- 10.3.4.3 (Appendix 7.6)

Description: This section was modified to include the condition where the steam turbine resource is online -- the logic where the PSU is deemed unavailable if the CT's maximum constraint is below it's MLP no longer applies. Instead, the real-time engine will calculate a PSUMax based on the CT and ST respective high operating limits below MLP.

Modified section:
- 10.3.4.4 (Appendix 7.6)

Description: This section was modified to include the condition when the ST is offline.

Added section:
- 5.6.1.6 (Appendix 7.6)

Description: This section was added to incorporate logic which determines the initial energy schedule based on the latest dispatch instruction when operator intervention has caused deviations from the original dispatch schedule.

Added section:
- 8.6.2.3 (Appendix 7.6)

Description: This section was added to incorporate logic for if a NQS's minimum constraint, maximum constraint, or derate is below the MLP. The RT calculation engine will respect that constraint and ramp rates in the dispatch and advisory intervals until that constraint or de-rate is removed or has ended.

Part 4 - Proposed Amendment

Appendix 7.5 -The Day-Ahead Market Calculation Engine Process

---

22 Pseudo-Unit Modelling

22.2 Application of Physical Resource Deratings to the Pseudo-Unit Model

22.2.1 The *day-ahead market calculation engine* shall apply deratings submitted by *market participants* to the applicable *dispatchable* capacity and duct firing capacity parameters for a *pseudo-unit*, where:

22.2.1.1 $CTCap_{h,k}$ designates the capacity of combustion turbine *resource* $k \in \{1,..,\mathrm{K}\}$ in hour h as determined by submitted deratings;

22.2.1.2 $STCap_h$ designates the capacity of the steam turbine *resource* in hour h as determined by submitted deratings; and

22.2.1.3 $TotalQ_{h,k}$ designates the total *offered* quantity of *energy* for *pseudo-unit* $k \in \{1,..,\mathrm{K}\}$ in hour h.

22.2.2 The *day-ahead market calculation engine* shall solve for the following operating region parameters for hour $h \in [1,..,24\}$ for each *pseudo-unit* $k \in \{1,..,\mathrm{K}\}$:

22.2.2.1 $MLP_{h,k}$ designates the *minimum loading point* of *pseudo-unit* k in hour h;

22.2.2.2 $DR_{h,k}$ designates the *dispatchable* region capacity of *pseudo-unit* k in hour h; and

22.2.2.3 $DF_{h,k}$ designates the duct firing region capacity of *pseudo-unit* k in hour h.

22.2.2A The *day-ahead market calculation engine* shall prorate the derating submitted by a *market participant* on the steam turbine *resource* in hour $h \in \{1,..,24\}$ to the associated *pseudo-units* when *pseudo-unit* $k \in \{1,..,\mathrm{K}\}$ for hour h meets all the following conditions:

22.2.2A.1 An *offer* for *pseudo-unit* k was submitted in hour h;

22.2.2A.2 *Pseudo-unit* k has no *planned outage* or *forced outage* on the associated combustion turbine *resource* in hour h;

22.2.2A.3 *Pseudo-unit* k has no fixed or maximum constraint on the associated combustion turbine *resource* below the designated *minimum loading point* of combustion turbine *resource* in hour h;

22.2.2A.4 *Pseudo-unit* k has no fixed or maximum constraint below the *minimum loading point* of *pseudo-unit* k in hour h; and

22.2.2A.5 *Pseudo-unit* k is not operating in *single cycle mode* for the day.

22.2.3 Pre-processing of De-rates

22.2.3.1 The *day-ahead market calculation engine* shall perform the following pre-processing steps to determine the available operating regions for a *pseudo-unit* based on the combustion turbine *resource* and steam turbine *resource* share and the application of the *pseudo-unit* deratings. For *pseudo-unit* $k \in \{1, . . ,K\}$ for hour $h \in \{1, . . , 24\}$:

22.2.3.1.1 Step 1: Calculate the amount of *offered energy* attributed to each combustion turbine *resource* ($CTAmt_{h,k}$) and steam turbine *resource* portion ($STAmt_{h,k}$):

If $TotalQ_{h,k} < MMLP_k$ then:

$CTAmt_{h,k} = 0$; and

$STAmt_{h,k} = 0$.

Otherwise:

$CTAmtMLP = MMLP_k \cdot CTShareMLP_k$; and

$STAmtMLP = (1 - CSCM_k) \cdot MMLP_k \cdot STShareMLP_k$.

If $TotalQ_{h,k} > MMLP_k + MDR_k$, then:

$CTAmtDR = MDR_k \cdot CTShareDR_k$;

$STAmtDR = (1 - CSCM_k) \cdot MDR_k \cdot STShareDR_k$; and

$STAmtDF = (1 - CSCM_k) \cdot (TotalQ_{h,k} - MMLP_k - MDR_k)$.

Otherwise:

$CTAmtDR = (TotalQ_{h,k} - MMLP_k) \cdot CTShareDR_k$;

$STAmtDR = (1 - CSCM_k) \cdot (TotalQ_{h,k} - MMLP_k) \cdot STShareDR_k$; and

$STAmtDF = 0$;

$CTAmt_{h,k} = CTAmtMLP + CTAmtDR$; and

$STAmt_{h,k} = STAmtMLP + STAmtDR + STAmtDF$.

22.2.3.1.2 Step 2: Allocate the steam turbine *resource* capacity ($PRSTCap_{h,k}$) to each steam turbine *resource* portion associated with *pseudo-unit* that meets all the conditions in Section 22.2.2A:

$$PRSTCap_{h,k} = \left(\frac{STAmt_{h,k}}{\sum_{w \in \{1,..,K\}} STAmt_{h,w}} \right) \cdot STCap_h$$

If *pseudo-unit* $k \in \{1, . . , \mathrm{K}\}$ in hour h does not meet all conditions in Section 22.2.2A, then:

$$STAmt_{h,k} = 0;$$ and

$$PRSTCap_{h,k} = 0.$$

22.3 Convert Physical Resource Constraints to Pseudo-Unit Constraints

---

22.3.5.2 If *pseudo-units* with sufficient steam turbine *resource* capacity are not committed, then the *day-ahead market calculation engine* shall not convert the entire quantity of the steam turbine *resource* minimum constraint to *pseudo-unit* constraints.

22.3.5.2A The *day-ahead market calculation engine* shall prorate a steam turbine *resource* maximum constraint in hour $h \in \{1, . . , 24\}$ to the associated *pseudo-units* when *pseudo-unit* $k \in \{1, . . , \mathrm{K}\}$ for hour h meets all the following conditions:

22.3.5.2A.1 An *offer* for *pseudo-unit* k was submitted in hour h;

22.3.5.2A.2 *Pseudo-unit* k has no *planned outage* or *forced outage* on the associated combustion turbine *resource* in hour h;

22.3.5.2A.3 *Pseudo-unit* k has no fixed or maximum constraint on the associated combustion turbine *resource* below the designated *minimum loading point* of combustion turbine *resource* in hour h;

22.3.5.2A.4 *Pseudo-unit* k has no fixed or maximum constraint below the *minimum loading point* of *pseudo-unit* k in hour h; and

22.3.5.2A.5 *Pseudo-unit* k is not operating in *single cycle mode* for the day.

22.3.5.3 The steam turbine *resource* maximum constraint shall be converted to a *pseudo-unit* constraint as follows:

The steam turbine *resource* portion $STAmt_{h,k}$ shall be calculated as per Section 22.2.3.1.1.

Allocate the steam turbine *resource* maximum constraints to each *pseudo-unit* that met the conditions in Section 22.3.5.2A as follows:

$$PRSTMax_{h,k} = \left(\frac{STAmt_{h,k}}{\sum_{w\in\{1,..,K\}} STAmt_{h,w}}\right) \cdot STMax$$

If the prorated steam turbine maximum constraint limits the steam turbine portion to below its *minimum loading point*, then

$$PSUMax_{h,k} = 0$$

Otherwise, calculate R so that $SMLP + R \cdot STShareDR_k \cdot MDR_k = PRSTMax_{h,k}$

If $R \leq 1$, set $PSUMax_{h,k} = MLP_{h,k} + min(DR_{h,k}, R \cdot MDR_k)$

If $R > 1$, set $PSUMax_{h,k} = MLP_{h,k} + DR_{h,k} + PRSTMax_{h,k} - SMLP - STShareDR_k \cdot MDR_k$

22.3.5.4 If the steam turbine *resource* minimum and maximum constraints are equal but do not convert to equal *pseudo-unit* minimum and maximum constraints, then the steam turbine *resource* minimum constraint conversion in section 22.3.5.1 shall be used to determine equal *pseudo-unit* minimum and maximum constraints.

Appendix 7.5A – The Pre-Dispatch Calculation Engine Process

---

15 Pseudo-Unit Modelling

15.2 Application of Physical Resource Deratings to the Pseudo-Unit Model

15.2.1 The *pre-dispatch calculation engine* shall apply deratings submitted by *market participants* to the applicable *dispatchable* capacity and duct firing capacity parameters for a *pseudo-unit*, where:

15.2.1.1 $CTCap_{t,k}$ designates the capacity of combustion turbine *resource* $k \in \{1,..,K\}$ in time-step t as determined by submitted deratings;

15.2.1.2 $STCap_t$ designates the capacity of the steam turbine *resource* in time-step t as determined by submitted deratings; and

15.2.1.3 $TotalQ_{t,k}$ designates the total quantity of *energy* for *pseudo-unit* $k \in \{1,..,K\}$ in time-step t.

15.2.2 The *pre-dispatch calculation engine* shall solve for the following operating region parameters for each *pseudo-unit* $k \in \{1,..,K\}$:

15.2.2.1 $MLP_{t,k}$, which designates the *minimum loading point* of *pseudo-unit k* in time-step t;

15.2.2.2 $DR_{t,k}$, which designates the *dispatchable* region capacity of *pseudo-unit* k in time-step t; and

15.2.2.3 $DF_{t,k}$, which designates the duct firing region capacity of *pseudo-unit k* in time-step t.

15.2.2A The *pre-dispatch calculation engine* shall prorate the derating submitted by a *market particpant* on the steam turbine *resource* in time-step $t \in TS$ to the associated *pseudo-units* when *pseudo-unit* $k \in \{1,..,K\}$ for time-step t meets all the following conditions:

15.2.2A.1 An *offer* for *pseudo-unit* k was submitted in time-step t;

15.2.2A.2 *Pseudo-unit* k has no *planned outage* or *forced outage* on the associated combustion turbine *resource* in time-step t;

15.2.2A.3 *Pseudo-unit* k has no fixed or maximum constraint on the associated combustion turbine *resource* below the designated *minimum loading point* of combustion turbine resource in time-step t;

15.2.2A.4 *Pseudo-unit* k has no fixed or maximum constraint below the *minimum loading point* of *pseudo-unit* k in time-step t; and

15.2.2A.5 *Pseudo-unit* k is not operating in *single cycle mode* for the day.

15.2.3 Pre-Processing of De-rates

15.2.3.1 The *pre-dispatch calculation engine* shall perform the following pre-processing steps to determine the available operating regions for a *pseudo-unit* based on the combustion turbine *resource's* and steam turbine *resource's* share and the application of the *pseudo-unit* deratings. For *pseudo-unit* $k \in \{1,..,K\}$ for time-step $t \in TS$:

15.2.3.1.1 Step 1: Calculate the amount of *offered energy* attributed to each combustion turbine *resource* ($CTAmt_{t,k}$) and steam turbine *resource* portion ($STAmt_{t,k}$):

If $TotalQ_{t,k} < MMLP_k$ then:

~~Calculate~~ $CTAmt_{t,k} = 0$; *and*

~~Calculate~~ $STAmt_{t,k} = 0$.

Otherwise:

$CTAmtMLP = MMLP_k \cdot CTShareMLP_k$; and

$STAmtMLP = (1 - CSCM_k) \cdot MMLP_k \cdot STShareMLP_k$.

If $TotalQ_{t,k} > MMLP_k + MDR_k$, then:

$CTAmtDR = MDR_k \cdot CTShareDR_k$;

$STAmtDR = (1 - CSCM_k) \cdot MDR_k \cdot STShareDR_k$; and

$STAmtDF = (1 - CSCM_k) \cdot (TotalQ_{t,k} - MMLP_k - MDR_k)$.

Otherwise:

$CTAmtDR = (TotalQ_{t,k} - MMLP_k) \cdot CTShareDR_k$;

$STAmtDR = (1 - CSCM_k) \cdot (TotalQ_{t,k} - MMLP_k) \cdot STShareDR_k$;

$STAmtDF = 0$;

$CTAmt_{t,k} = CTAmtMLP + CTAmtDR$; and

$STAmt_{t,k} = STAmtMLP + STAmtDR + STAmtDF$.

15.2.3.1.2 Step 2: Allocate the steam turbine *resource's* capacity to each steam turbine *resource* portion associated with a *pseudo-unit* that meets all conditions in section 15.2.2A:

$$PRSTCap_{t,k} = \left(\frac{STAmt_{t,k}}{\sum_{w \in \{1,..,K\}} STAmt_{t,w}}\right) \cdot STCap_t$$

If *pseudo-unit* $k \in \{1, .., K\}$ in time-step t does not meet all conditions in section 15.2.2A, then:

$STAmt_{t,k} = 0$; and

$PRSTCap_{t,k} = 0$.

15.3 Convert Physical Resource Constraints to Pseudo-Unit Constraints

---

15.3.5.2 If *pseudo-units* with sufficient steam turbine *resource* capacity are not commited, then the *pre-dispatch calculation engine* shall not convert the entire quantity of the steam turbine *resource's* minimum constraint to *pseudo-unit* constraints.

15.3.5.2A The *pre-dispatch calculation engine* shall prorate the steam turbine *resource*'s maximum constraint in time-step $t \in TS$ to the associated *pseudo-units* when *pseudo-unit* $k \in \{1, .., K\}$ for time-step t meets all the following conditions:

15.3.5.2A.1 An *offer* for *pseudo-unit* k was submitted in time-step t;

15.3.5.2A.2 *Pseudo-unit* k has no *planned outage* or *forced outage* on the associated combustion turbine *resource* in time-step t;

15.3.5.2A.3 *Pseudo-unit* k has no fixed or maximum constraint on the associated combustion turbine *resource* below the designated *minimum loading point* of combustion turbine *resource* in time-step t;

15.3.5.2A.4 *Pseudo-unit* k has no fixed or maximum constraint below the *minimum loading point* of *pseudo-unit* k in time-step t; and

15.3.5.2A.5 *Pseudo-unit* k is not operating in *single cycle mode* for the day.

15.3.5.3 The steam turbine *resource's* maximum constraint shall be converted to a *pseudo-unit* constraint as follows:

15.3.5.3.1 The steam turbine *resource* portion ($STAmt_{t,k}$) shall be calculated as per section 15.2.3.1.1.

15.3.5.3.2 Allocate the steam turbine *resource* maximum constraints to each *pseudo-unit* that met the conditions in section 15.3.5.2A as follows:

$$PRSTMax_{t,k} = \left(\frac{STAmt_{t,k}}{\sum_{w\in\{1,..,K\}} STAmt_{t,w}}\right) \cdot STMax.$$

If *pseudo-unit* $k \in \{1, . . ,K\}$ in time-step t does not meet all conditions in section 15.3.5.2A, then:

$STAmt_{t,k} = 0$; and

$PRSTCap_{t,k} = 0$.

15.3.5.3.3 If the converted steam turbine *resource* maximum constraint limits the steam turbine *resource* portion to below its *minimum loading point*, then

$$PSUMax_{t,k} = 0.$$

15.3.5.3.4 Otherwise, calculate R so that $SMLP + R \cdot STShareDR_k \cdot MDR_k = PRSTMax_{t,k}$:

If $R \leq 1$, set $PSUMax_{t,k} = MLP_{t,k} + min(DR_{t,k}, R \cdot MDR_k)$.

If $R > 1$, set $PSUMax_{t,k} = MLP_{t,k} + DR_{t,k} + PRSTMax_{t,k} - SMLP - STShareDR_k \cdot MDR_k$.

15.3.5.4 If the steam turbine *resource's* minimum and maximum constraints are equal but do not convert to equal *pseudo-unit* minimum and maximum constraints, then the steam turbine *resource* minimum constraint conversion in section 15.3.5.1 shall be used to determine equal *pseudo-unit* minimum and maximum constraints.

Appendix 7.6 – The Real-Time Calculation Engine Process

5.6 Initial Scheduling Assumptions

5.6.1 Initial Schedules

5.6.1.1 Initial *energy* schedules shall be based on the values determined by the *IESO's energy* management system and the schedules from the previous *real-time calculation engine* run, where:

5.6.1.1.1 $RTDLTel_{-1,b}$ designates the *energy* management system MW value for the *dispatchable load* at bus $b \in B^{DL}$;

5.6.1.1.2 $SDLSch_{0,b}^{Prev}$ designates the schedule determined for the *dispatchable load* at bus $b \in B^{DL}$ by the Real-Time Scheduling algorithm in section 8, of the previous *real-time calculation engine* run;

5.6.1.1.3 $RTDGTel_{-1,b}$ designates the *energy* management system MW value for the *dispatchable generation resource* at bus $b \in B^{DG}$;

5.6.1.1.4 $SDGSch_{0,b}^{Prev}$ designates the schedule determined for the *dispatchable generation resource* at bus $b \in B^{DG}$ by the Real-Time Scheduling algorithm in section 8, of the previous *real-time calculation engine* run;

5.6.1.1.5 $SDLPrc_{0,b}^{Prev}$ designates the schedule determined for the *dispatchable load* at bus $b \in B^{DL}$ by the Real-Time Pricing algorithm in section 9, of the previous *real-time calculation engine* run; and

5.6.1.1.6 $SDGPrc_{0,b}^{Prev}$ designates the schedule determined for the *dispatchable generation resource* at bus $b \in B^{DG}$ by the Real-Time Pricing algorithm in section 9, of the previous *real-time calculation engine* run.

5.6.1.2 For the *dispatchable load* at bus b, the intitial schedule, $SDLInitSch_{0,b}$, for the Real-Time Scheduling algorithm in section 8, shall be determined as follows:

5.6.1.2.1 Step 1: Calculate $TelUp_{0,b}$ using the submitted up ramp rates and break points to determine the maximum

consumption level the *dispatchable load* can achieve in five minutes from $RTDLTel_{-1,b}$;

5.6.1.2.2 Step 2: Calculate $TelDown_{0,b}$ using the submitted down ramp rates and break points to determine the minimum consumption level the *dispatchable load* can achieve in five minutes from $RTDLTel_{-1,b}$; and

5.6.1.2.3 Step 3: If the schedule from the previous *real-time calculation engine* run is achievable by ramping from the $RTDLTel_{-1,b}$, then set the initial schedule to the schedule from the previous *real-time calculation engine* run. Otherwise, set the initial schedule to the nearest boundary:

If $TelDown_{0,b} \leq SDLSch_{0,b}^{Prev} \leq TelUp_{0,b}$, then set $SDLInitSch_{0,b} = SDLSch_{0,b}^{Prev}$

If $SDLSch_{0,b}^{Prev} < TelDown_{0,b}$, then set $SDLInitSch_{0,b} = TelDown_{0,b}$

Otherwise, set $SDLInitSch_{0,b} = TelUp_{0,b}$.

5.6.1.3 For the *dispatchable generation resource* at bus b, the intitial schedule, $SDGInitSch_{0,b}$, for the Real-Time Scheduling algorithm in section 8, shall be determined as follows:

5.6.1.3.1 Step 1: Calculate $TelUp_{0,b}$ using the submitted up ramp rates and break points to determine the maximum production level the *resource* can achieve in five minutes from $RTDGTel_{-1,b}$;

5.6.1.3.2 Step 2: Calculate $TelDown_{0,b}$ using the submitted down ramp rates and break points to determine the minimum production level the *resource* can achieve in five minutes from $RTDGTel_{-1,b}$; and

5.6.1.3.3 Step 3: If the schedule from the previous *real-time calculation engine* run is achievable by ramping from the $RTDGTel_{-1,b}$, then set the initial schedule to the schedule from the previous *real-time calculation engine* run. Otherwise, set the initial schedule to the nearest boundary:

If $TelDown_{0,b} \leq SDGSch_{0,b}^{Prev} \leq TelUp_{0,b}$ then set $SDGInitSch_{0,b} = SDGSch_{0,b}^{Prev}$

If $SDGSch_{0,b}^{Prev} < TelDown_{0,b}$ then set $SDGInitSch_{0,b} = TelDown_{0,b}$

Otherwise, set $SDGInitSch_{0,b} = TelUp_{0,b}$.

5.6.1.4 For the *dispatchable load* at bus b, the intitial schedule, $SDLInitPrc_{0,b}$, for the Real-Time Pricing algorithm in section 9, shall be determined as follows:

If $SDLSch_{0,b}^{Prev} \leq SDLPrc_{0,b}^{Prev} \leq SDLInitSch_{0,b}$ or $SDLInitSch_{0,b} \leq SDLPrc_{0,b}^{Prev} \leq SDLSch_{0,b}^{Prev}$, then set $SDLInitPrc_{0,b} = SDLInitSch_{0,b}$;

Otherwise set $SDLInitPrc_{0,b} = SDLPrc_{0,b}^{Prev}$.

5.6.1.5 For the *dispatchable generation* at bus *b*, the intitial schedule $SDGInitPrc_{0,b}$, for the Real-Time Pricing algorithm in section 9, designates the initial schedule for the *dispatchable generation resource* at bus *b* and is determined as follows:

If $SDGSch_{0,b}^{Prev} \leq SDGPrc_{0,b}^{Prev} \leq SDGInitSch_{0,b}$ or $SDGInitSch_{0,b} \leq SDGPrc_{0,b}^{Prev} \leq SDGSch_{0,b}^{Prev}$ then set $SDGInitPrc_{0,b} = SDGInitSch_{0,b}$;

Otherwise set $SDGInitPrc_{0,b} = SDGPrc_{0,b}^{Prev}$.

5.6.1.6 Where *dispatch instructions* are issued by the *IESO* due to a manual constraint, the *real-time calculation engine* shall determine the initial *energy* schedule using the most recent *dispatch instructions* in place of $SDLSch_{0,b}^{Prev}$ and $SDGSch_{0,b}^{Prev}$.

---

8 Real-Time Scheduling

8.6 Dispatch Data Inter-Interval/Multi-Interval Constraints

8.6.2 Non-Quick Start Resource Start-up and Shutdown

8.6.2.1 For all intervals in the real-time look-ahead period in which a *non-quick start resource* is scheduled to start-up, such *resource* shall be scheduled on a fixed ramp-up trajectory as determined by its *offered* ramp rates. The ramp-up trajectory ($UpTraj_{i,b}$) for interval $i \in I$ such that $SU_{i,b}=1$ is determined as follows:

8.6.2.1.1 If $i = 1$, then $UpTraj_{i,b}$ shall be determined from the *resource* initial schedule and the *offered* ramp up capability;

8.6.2.1.2 If $i > 1$ and $SU_{i\text{-}1,b} = 0$, then $UpTraj_{i,b}$ shall be determined from the *offered* ramp up capability from 0; and

8.6.2.1.3 For all intervals $i \in I$ such that $SU_{i,b}=1$:

$$\sum_{k \in K^{E}_{i,b}} SDG_{i,b,k} = UpTraj_{i,b}.$$

8.6.2.2 For all intervals in the real-time look-ahead period in which a *non-quick start resource* is scheduled to shutdown, such *resource* shall be scheduled on a fixed ramp-down trajectory as determined by its *offered* ramp rates. The ramp-down trajectory ($DnTraj_{i,b}$) for interval $i \in I$ such that $SD_{i,b} = 1$ is determined as follows:

8.6.2.2.1 If $i = 1$, then $DnTraj_{i,b}$ shall be determined from the *resource* initial schedule and the *offered* ramp down capability;

8.6.2.2.2 If $i > 1$ and $SD_{i\text{-}1,b} = 0$, then $DnTraj_{i,b}$ shall be $MinQDG_b$; and

8.6.2.2.3 If $i > 1$ and $SD_{i\text{-}1,b} = 1$, then $DnTraj_{i,b}$ shall be determined from the *offered* ramp down capability from $DnTraj_{i\text{-}1,b}$.

8.6.2.2.4 For all intervals $i \in I$ such that $SD_{i,b} = 1$:

$$\sum_{k \in K^{E}_{i,b}} SDG_{i,b,k} = DnTraj_{i,b}.$$

8.6.2.3 Where a *resource*'s minimum constraint, maximum constraint, or derating below *minimum loading point* overlaps with the calculated ramping trajectory of a *non-quick start* resource, the *real-time calculation engine* shall enforce the applicable constraint or derating for the duration of that overlap. Upon the expiry or removal of the constraint or derating, the *real-time calculation engine* shall continue ramping the *non-quick start resource* from the most recently established quantity in the *real-time schedule*.

10 Pseudo-Unit Modelling

10.2 Application of Physical Resource Deratings to the Pseudo-Unit Model

10.2.1 The *real-time calculation engine* shall apply deratings submitted by *market participants* to the applicable *minimum loading point*, *dispatchable* capacity and duct firing capacity parameters for a *pseudo-unit*, where:

10.2.1.1 $CTCap_{i,k}$ designates the capacity of combustion turbine *resource* $k \in \{1,..,K\}$ in interval i as determined by submitted deratings;

10.2.1.2 $STCap_i$ designates the capacity of the steam turbine *resource* in interval i as determined by submitted deratings; and

10.2.1.3 $TotalQ_{i,k}$ designates the total *offered* quantity of *energy* for *pseudo-unit* $k \in \{1,..,K\}$ in interval i.

10.2.2 The *real-time calculation engine* shall solve for the following operating region parameters for each *pseudo-unit* $k \in \{1,..,K\}$:

10.2.2.1 $MLP_{i,k}$ designates the *minimum loading point* of *pseudo-unit* k in interval i;

10.2.2.2 $DR_{i,k}$ designates the *dispatchable* capacity region of *pseudo-unit* k in interval i; and

10.2.2.3 $DF_{i,k}$ designates the duct firing capacity region of *pseudo-unit* k in interval i.

10.2.2A The *real-time calculation engine* shall prorate the derating submitted by the *market participant* on the steam turbine *resource* in interval $i \in I$ to the associated *pseudo-units* when *pseudo-unit* $k \in \{1, . . ,K\}$ for interval i meets all the following conditions:

10.2.2A.1 An *offer* for *pseudo-unit* k was submitted in interval i;

10.2.2A.2 *Pseudo-unit* k has no *planned outage* or *forced outage* on the associated combustion turbine *resource* in interval i;

10.2.2A.3 *Pseudo-unit* k has no 0MW fixed or maximum constraint on the associated combustion turbine *resource* in interval i;

10.2.2A.4 *Pseudo-unit* k has no 0MW fixed or maximum constraint in interval i; and

10.2.2A.5 *Pseudo-unit* k is not operating in *single cycle mode* for the day.

10.2.3 Pre-Processing of De-rates

10.2.3.1 The *real-time calculation engine* shall perform the following pre-processing steps to determine the available operating regions for a *pseudo-unit* based on the combustion turbine *resource's* and steam turbine *resource's* share and the application of the *pseudo-unit* deratings. For *pseudo-unit* $k \in \{1,..,K\}$ for interval $i \in I$:

10.2.3.1.1 Step 1: Calculate the amount of the *offer* for *energy* that is attributed to each combustion turbine *resource* ($CTAmt_{i,k}$) and steam turbine *resource* portion ($STAmt_{i,k}$):

If $TotalQ_{i,k} < MMLP_k$, then:

~~Calculate~~ $CTAmt_{i,k} = 0$; *and*

~~Calculate~~ $STAmt_{i,k} = 0$.

Otherwise:

$CTAmtMLP = MMLP_k \cdot CTShareMLP_k$; and

$STAmtMLP = (1 - CSCM_k) \cdot MMLP_k \cdot STShareMLP_k$.

If $TotalQ_{i,k} > MMLP_k + MDR_k$, then:

$CTAmtDR = MDR_k \cdot CTShareDR_k$;

$STAmtDR = (1 - CSCM_k) \cdot MDR_k \cdot STShareDR_k$; and

$STAmtDF = (1 - CSCM_k) \cdot (TotalQ_{i,k} - MMLP_k - MDR_k)$.

Otherwise:

$CTAmtDR = (TotalQ_{i,k} - MMLP_k) \cdot CTShareDR_k$;

$STAmtDR = (1 - CSCM_k) \cdot (TotalQ_{i,k} - MMLP_k) \cdot STShareDR_k$;

$STAmtDF = 0$;

$CTAmt_{i,k} = CTAmtMLP + CTAmtDR$; and

$STAmt_{i,k} = STAmtMLP + STAmtDR + STAmtDF$.

10.2.3.1.2 Step 2: Allocate the steam turbine *resource's* capacity to each steam turbine *resource* portion associated with the *pseudo-unit* that meets all conditions in section 10.2.2A:

$$PRSTCap_{i,k} = \left(\frac{STAmt_{i,k}}{\sum_{w \in \{1,..,K\}} STAmt_{i,w}}\right) \cdot STCap_i$$

If *pseudo-unit* $k \in \{1, . . , K\}$ in interval i does not meet all conditions in section 10.2.2A, then:

$STAmt_{i,k} = 0$; and

$PRSTCap_{i,k} = 0.$

10.2.3.1.3 Step 3: Determine if the *pseudo-unit* is available:

If $CTAmt_{i,k} < CMLP_k$, then the *pseudo-unit* is unavailable.

If $STAmt_{i,k} < SMLP \cdot (1 - CSCM_k)$, then the *pseudo-unit* is unavailable.

~~If $CTCap_{i,k} < CMLP_k$, then the *pseudo-unit* is unavailable.~~

~~If $PRSTCap_{i,k} < SMLP \cdot (1 - CSCM_k)$, then the *pseudo-unit* is unavailable.~~

10.2.3.1.4 Step 4: Initialize the operating region parameters for interval $i \in I$ to the model parameter values:

Set $MLP_{i,k} = MMLP_k$.

Set $DR_{i,k} = MDR_k$.

Set $DF_{i,k} = MDF_k$.

10.2.3.1.5 Step 5: Apply the derating for the combustion turbine *resource* to the *dispatchable* and *minimum loading point* regions for *pseudo-units* not operating in *single cycle mode* where the associated steam turbine *resource* is online:

If $CTCap_{i,k} < CMLP_k$, then

Calculate $PRSTCapMLP_{i,k} = CTCap_{i,k} \cdot \left(\frac{STShareMLP_k}{CTShareMLP_k}\right)$

Update $MLP_{i,k} = CTCap_{i,k} + PRSTCapMLP_{i,k}$;

$DR_{i,k} = 0$; and $DF_{i,k} = 0.$

Otherwise:

Calculate P so that $CMLP_k + P \cdot CTShareDR_k \cdot MDR_k = CTCap_{i,k}$; and

~~Set~~ Update $DR_{i,k} = min(DR_{i,k}, P \cdot MDR_k)$.

10.2.3.1.5A If a *pseudo-unit* is flagged to operate in *single cycle mode* or the associated steam turbine *resource* is offline, then the combustion

turbine *resource's* derating shall be applied to the *dispatchable* and *minimum loading point* regions of the *pseudo-unit* as follows:

If $CTCap_{i,k} < CMLP_k$, then

Update $MLP_{i,k} = CTCap_{i,k}$; $DR_{i,k} = 0$, and $DF_{i,k} = 0$.

Otherwise:

Calculate P so that $CMLP_k + P \cdot CTShareDR_k \cdot MDR_k = CTCap_{i,k}$; and

Update $MLP_{i,k} = CMLP_k$; $DR_{i,k} = min(DR_{i,k}, P \cdot MDR_k)$; and $DF_{i,k} = 0$.

10.2.3.1.6 Step 6: Apply the derating for the steam turbine *resource* to the duct firing ~~, and~~ *dispatchable* *capacity*, and *minimum loading point* regions for *pseudo-units* that meets all conditions in section 10.2.2A~~not operating in *single cycle mode*~~:

If $PRSTCap_{i,k} < SMLP \cdot (1 - CSCM_k)$, then

Update $MLP_{i,k} = CMLP_k + PRSTCap_{i,k}$;
$DR_{i,k} = 0$; and $DF_{i,k} = 0$.

Otherwise:

Calculate R so that $SMLP + R \cdot STShareDR_k \cdot MDR_k = PRSTCap_{i,k}$.

If $R \leq 1$, update $DF_{i,k} = 0$, and $DR_{i,k} = min(DR_{i,k}, R \cdot MDR_k)$.

If $R > 1$, update $DF_{i,k} = min(DF_{i,k}, PRSTCap_{i,k} - SMLP - STShareDR_k \cdot MDR_k)$.

10.2.4 Available Energy Laminations

10.2.4.1 The *real-time calculation engine* shall determine the *offer* quantity laminations that may be scheduled for *energy* and *operating reserve* in each operating region for interval $i \in I$ for each *pseudo-unit* $k \in \{1,..,K\}$, subject to section 10.2.4.2, where:

10.2.4.1.1 $QMLP_{i,k}$ designates the total quantity that may be scheduled in the *minimum loading point* region;

10.2.4.1.2 $QDR_{i,k}$ designates the total quantity that may be scheduled in the *dispatchable* region; and

10.2.4.1.3 $QDF_{i,k}$ designates the total quantity that may be scheduled in the duct firing region.

10.2.4.2 The available *offered* quantity laminations shall be subject to the following conditions:

$$0 \leq QMLP_{i,k} \leq MLP_{i,k};$$

$$0 \leq QDR_{i,k} \leq DR_{i,k};$$

$$0 \leq QDF_{i,k} \leq DF_{i,k};$$

if $QMLP_{i,k} < MLP_{i,k}$, then the $pseudo-unit$ is unavailable and $QDR_{i,k} = QDF_{i,k} = 0$; and

if $QDR_{i,k} < DR_{i,k}$, then $QDF_{i,k} = 0$.

10.3 Convert Physical Resource Constraints to Pseudo-Unit Constraints

10.3.1 The *real-time calculation engine* shall convert physical *resource* constraints to *pseudo-unit* constraints, where:

10.3.1.1 $PSUMin^{q}_{i,k}$ designates the minimum limitation on *pseudo-unit* k determined by translating constraint q. When constraint *q* does not provide a minimum limitation on *pseudo-unit* k, then $PSUMin^{q}_{i,k}$ shall be set equal to 0;

10.3.1.2 $PSUMax^{q}_{i,k}$ designates the maximum limitation on *pseudo-unit* k determined by translating constraint q. When constraint *q* does not provide a maximum limitation on *pseudo-unit* k, then $PSUMax^{q}_{i,k}$ shall be set equal to $MLP_{i,k}+DR_{i,k}+DF_{i,k}$; and

10.3.1.3 $CTCmtd_{i,k} \in \{0,1\}$ designates whether combustion turbine *resource* $k \in \{1,..K\}$ is considered committed in interval $i \in I$.

10.3.2 The *real-time calculation engine* shall calculate the minimum and maximum limitations, subject to section 10.3.3.1, as follows:

10.3.2.1 Minimum limitation: $MinDG_{i,k} = \max_{q\in\{1,..Q\}} PSUMin^{q}_{i,k}$

10.3.2.2 Maximum limitation: $MaxDG_{i,k} = \min_{q\in\{1,..Q\}} PSUMax^{q}_{i,k}$

where Q designates the number of constraints impacting a *combined cycle plant* that have been provided to the *real-time calculation engine*.

10.3.3 Pseudo-Unit Minimum and Maximum Constraints

10.3.3.1 *Pseudo-unit* minimum and maximum constraints shall be calculated as follows:

10.3.3.1.1 $PSUMin_{i,k} = PMin$ where $PMin$ shall be a minimum constraint provided on *pseudo-unit* $k \in \{1,..,K\}$ for interval $i \in I$; and

10.3.3.1.2 $PSUMax_{i,k} = PMax$ where $PMax$ shall be a maximum constraint provided on *pseudo-unit* $k \in \{1,..,K\}$ for interval $i \in I$.

10.3.3.2 Where *pseudo-unit* $k \in \{1,..,K\}$ for interval $i \in I$ is not flagged to operate in *single cycle mode* and either the maximum constraint or maximum *offered* quantity is below the *minimum loading point*, the *real-time calculation engine* shall adjust the the maximum constraints on the associated combustion turbine *resource* $(CTMax_{i,k})$ and steam turbine *resource* portion $(PRSTMax_{i,k})$ in accordance with the following:

10.3.3.2.1 Where the associated combustion turbine *resource* is online and the steam turbine *resource* is offline, the combustion turbine *resource* maximum constraint and steam turbine *resource* portion maximum constraint shall be determined as follows:

$CTMax_{i,k} = PSUMax_{i,k} * CTShareMLP_k$; and
$PRSTMax_{i,k} = 0$.

10.3.3.2.2 Where both the combustion turbine *resource* and steam turbine *resource* are online, the maximum constraints shall be determined as follows:

$CTMax_{i,k} = PSUMax_{i,k} * CTShareMLP_k$; and
$PRSTMax_{i,k} = PSUMax_{i,k} - CTMax_{i,k}$.

The aggregate steam turbine maximum constraint shall equal the summation of the steam turbine *resource* portion maximum constraints across all applicable *pseudo-units*.

10.3.3.2.3 Where the associated combustion turbine *resource* and/or steam turbine *resource* are not online and a future *start-up time* has been confirmed, the *real-time calculation engine* shall apply the calculated maximum constraints determined in Sections 10.3.3.2.1 and 10.3.3.2.2 to the advisory intervals, using the confirmed *start-up time* to determine when the associated *resource* is considered online.

10.3.4 Combustion Turbine Resource Minimum and Maximum Constraints

10.3.4.1 If the *pseudo-unit* is not flagged to operate in *single cycle mode* and the associated steam turbine *resource* is online, then the combustion turbine *resource's* minimum constraint shall be converted to a *pseudo-unit* constraint as follows:

If $CTMin < MLP_{i,k} \cdot CTShareMLP_k$, then set

$$STMinMLP = CTMin \cdot \left(\frac{STShareMLP_k}{CTShareMLP_k}\right),$$

$$STMinDR = 0.$$

Otherwise, if $CTMin \geq MLP_{i,k} \cdot CTShareMLP_k$, then set

$$STMinMLP = MLP_{i,k} \cdot STShareMLP_k,$$

$$STMinDR = \left(CTMin - MLP_{i,k} \cdot CTShareMLP_k\right) \cdot \left(\frac{STShareDR_k}{CTShareDR_k}\right).$$

Therefore:

$$PSUMin_{i,k} = CTMin + STMinMLP + STMinDR.$$

10.3.4.2 If a *pseudo-unit* is flagged to operate in *single cycle mode* or the associated steam turbine *resource* is offline, then the combustion turbine *resource's* minimum constraint shall be converted to a *pseudo-unit* constraint as follows:

$$PSUMin_{i,k} = CTMin$$

10.3.4.3 If the *pseudo-unit* is not flagged to operate in *single cycle mode* and the associated steam turbine *resource* is online, then the combustion turbine *resource's* maximum constraint shall be converted to a *pseudo-unit* constraint as follows:

If $CTMax < MLP_{i,k} \cdot CTShareMLP_k$, then ~~$PSUMax_{i,k} = 0$ and the *pseudo-unit* is unavailable.~~

$$PRSTMaxMLP = CTMax \cdot \left(\frac{STShareMLP_k}{CTShareMLP_k}\right)$$

$$PSUMax_{i,k} = CTMax + PRSTMaxMLP$$

Otherwise, calculate the value of the constraint on the steam turbine *resource* portion within the *minimum loading point* and *dispatchable* regions:

$$PRSTMaxMLP = MLP_{i,k} \cdot STShareMLP_k$$

~~$$STMaxDR = \left(CTMax - MLP_{i,k} \cdot CTShareMLP_k\right) \left(\frac{STShareDR_k}{CTShareDR_k}\right)$$~~

$$PRSTMaxDR = \left(CTMax - MLP_{i,k} \cdot CTShareMLP_k\right) \cdot \left(\frac{STShareDR_k}{CTShareDR_k}\right)$$

$$PSUMax_{i,k} = CTMax + PRSTMaxMLP + PRSTMaxDR$$

10.3.4.4 If a *pseudo-unit* is flagged to operate in *single cycle mode* or the associated steam turbine *resource* is offline, then the combustion turbine *resource's* maximum constraint shall be converted to a *pseudo-unit* constraint as follows:

$$PSUMax_{i,k} = CTMax.$$

10.3.5 Steam Turbine Resource Minimum and Maximum Constraints

10.3.5.1 The *real-time calculation engine* shall convert a stream turbine *resource's* minimum constraint to a *pseudo-unit* constraints as follows:

10.3.5.1.1 Step 1: Identify $A \subseteq \{1,..,K\}$, which designates the set of *pseudo-units* to which the constraint may be allocated where *pseudo-unit* $k \in \{1,..,K\}$ is placed in set A if and only if $CSCM_k = 0$ and $CTCmtd_{i,k} = 1$. If the set A is empty, then no further steps are required, otherwise proceed to Step 2.

10.3.5.1.2 Step 2: Determine the steam turbine *resource's* portion of the capacity for *pseudo-unit* $k \in A$:

$$STCap_k = QMLP_{i,k} \cdot STShareMLP_k + QDR_{i,k} \cdot STShareDR_k + QDF_{i,k}.$$

10.3.5.1.3 Step 3: Allocate the $STMin$ constraint to each *pseudo-unit* $k \in A$, where $STMin$ constraint shall be allocated equally to each *pseudo-unit* $k \in A$, where $STPMin_k$ is limited by $STCap_k$.

10.3.5.1.4 Step 4: The steam turbine *resource* portion minimum constraint shall be converted to a *pseudo-unit* constraint, where for each *pseudo-unit* $k \in A$:

If $STPMin_k < MLP_{i,k} \cdot STShareMLP_k$, then set

$CTMinMLP_k = STPMin_k \cdot \left(\frac{CTShareMLP_k}{STShareMLP_k}\right)$; and

$CTMinDR_k = 0$.

Otherwise, if $STPMin_k \geq MLP_{i,k} \cdot STShareMLP_k$, then set

$CTMinMLP_k = MLP_{i,k} \cdot CTShareMLP_k$; and

$CTMinDR_k = \left(STPMin_k - MLP_{i,k} \cdot STShareMLP_k\right) \cdot \left(\frac{CTShareDR_k}{STShareDR_k}\right)$.

Therefore:

$$PSUMin_{i,k} = STPMin_k + CTMinMLP_k + CTMinDR_k.$$

10.3.5.1.4 If *pseudo-units* with sufficient steam turbine *resource* capacity are not commited, then the *real-time calculation engine* shall not convert the entire quantity of the steam turbine *resource's* minimum constraint to *pseudo-unit* constraints.

10.3.5.1A The *real-time calculation engine* shall prorate maximum constraints submitted by the *market participant* for the steam turbine *resource* in interval $i \in I$ to the associated *pseudo-units* when *pseudo-unit* $k \in \{1, .., K\}$ for interval i meets all the following conditions:

10.3.5.1A.1 An *offer* for *pseudo-unit* k was submitted in interval i;

10.3.5.1A.2 *Pseudo-unit* k has no- *planned outage* or *forced outage* submitted on the associated combustion turbine *resource* in interval i;

10.3.5.1A.3 *Pseudo-unit* k has no 0MW fixed or maximum constraint on the associated combustion turbine *resource* in interval i;

10.3.5.1A.4 *Pseudo-unit* k has no 0MW fixed or maximum constraint in interval i; and

10.3.5.1A.5 *Pseudo-unit* k is not operating in *single cycle mode* for the day.

10.3.5.2 The steam turbine *resource's* maximum constraint shall be converted to a *pseudo-unit* constraint as follows:

10.3.5.2.1 The steam turbine *resource* portion $STAmt_{i,k}$ shall be calculated as per section 10.2.3.1.1.

10.3.5.2.2 Allocate the steam turbine *resource* maximum constraints to each *pseudo-unit* that met the conditions in section 10.3.5.1A as follows:

$$PRSTMax_{i,k} = \left(\frac{STAmt_{i,k}}{\sum_{w\in\{1,..,K\}} STAmt_{i,w}}\right) \cdot STMax.$$

10.3.5.2.3 If the converted steam turbine *resource* maximum constraint limits the steam turbine *resource* portion to below its *minimum loading point*, then

Set $PSUMax_{i,k} = CMLP_k + PRSTMax_{i,k}$

~~0~~.

10.3.5.2.4 Otherwise, calculate *R* so that $SMLP + R \cdot STShareDR_k \cdot MDR_k = PRSTMax_{i,k}$

If $R \leq 1$, set $PSUMax_{i,k} = MLP_{i,k} + min(DR_{i,k}, R \cdot MDR_k)$.

If $R > 1$, set $PSUMax_{i,k} = MLP_{i,k} + DR_{i,k} + PRSTMax_{i,k} - SMLP - STShareDR_k \cdot MDR_k$.

10.3.5.3 If the steam turbine *resource's* minimum and maximum constraints are equal but do not convert to equal *pseudo-unit* minimum and maximum constraints, then the steam turbine *resource* minimum constraint conversion in section 10.3.5.1 shall be used to determine equal *pseudo-unit* minimum and maximum constraints.
