วิธีการควบคุมการสลับโหมด GFL/GFM ระดับระบบ

Event-Triggered Finite-Control-Set Predictive Supervisor (ET-FCSPS)

แนวทางทางเทคนิคสำหรับ IEEE 14-bus: 1 SG + 4 IBR

เอกสารข้อเสนอวิธีวิจัยและแผนพัฒนา

4 สิงหาคม 2569

บทคัดย่อ

ระบบปัจจุบันใช้ AGSI++ เป็นดัชนีความเครียดราย IBR ร่วมกับ hysteresis และ dwell timer ในการเปลี่ยนโหมด GFL/GFM วิธีนี้ตรวจจับความผิดปกติเฉพาะจุดได้ แต่ยังไม่ได้ตอบคำถามระดับระบบว่า ควรเปลี่ยน IBR ตัวใด จำนวนเท่าใด และตัวใดควรเป็น angle/frequency reference leader เพื่อให้สภาวะของโครงข่ายทั้งระบบเหมาะสมก่อนสั่งสลับจริง รายงานนี้เสนอ ET-FCSPS ซึ่งใช้ AGSI++ เป็น event indicator แล้วแจกแจง binary mode vector ของ IBR ทั้งสี่ตัว ตรวจ hard constraints จาก full-network KCL และ reserve contract ทำนายผลของ candidate ที่ผ่านในช่วงเวลาสั้น จากนั้นเลือกคำสั่งด้วย objective และ tie-break ที่กำหนดไว้ล่วงหน้า

เอกสารฉบับนี้อธิบายสมการ ตัวแปร ลำดับข้อมูล candidate isolation, feasibility gate, cost function, reference transaction, fail-closed fallback, computational complexity, pseudocode, interface และแผนทดสอบโดยละเอียด วิธีที่เสนอยังไม่ได้ implement และยังไม่มีผล closed-loop จึงกล่าวเฉพาะประโยชน์ที่คาดหวังในรูป testable hypotheses เท่านั้น ไม่เพิ่ม AVR, ไม่ใช้ LLM หรือ BO ในวงรอบสั่งสลับ และไม่เปลี่ยนสมการ SG/IBR ที่ผ่านการตรวจแล้ว

สารบัญ

1 สถานะของเอกสารและขอบเขตหลักฐาน 3

2 ปัญหาของ AGSI++ แบบ local-only 3
- 2.1 สมการที่ใช้อยู่ปัจจุบัน . . . 3
- 2.2 AGSI++ มองเห็นอะไรและยังไม่เห็นอะไร . . . 4

3 แนวคิดหลักของ ET-FCSPS 4

4 นิยามข้อมูลเข้าและ system snapshot 4

5 decision variables และ candidate universe 5

6 event trigger และ candidate generation 5

7 hard feasibility gates 6
- 7.1 algebraic screening . . . 6
- 7.2 สิ่งที่ยังไม่ใช่ feasibility proof . . . 6

8 short-horizon prediction 7

9 objective function และ normalization 7
- 9.1 กติกาการกำหนด weight . . . 8

10 deterministic ranking และ tie-break 8

11 reference leader และ transaction semantics 8
11.1 SG trip หรือ reference loss 8
11.2 SG reclose และ handback 8

12 fail-closed fallback 9

13 pseudocode ของ controller 9

14 computational complexity และการลด runtime 9

15 ผลลัพธ์ที่คาดหวังและวิธีพิสูจน์ 10

16 การเปรียบเทียบกับ Bayesian optimisation 10

17 แผน interface สำหรับ implementation 11

18 แผนพัฒนาเป็นเฟส 11

19 verification matrix 11

20 ตัวอย่างลำดับการตัดสินใจ 12
20.1 SG trip 12
20.2 fault เฉพาะจุด 12
20.3 SG reclose 12

21 แนวตอบคำถามทางเทคนิค 12

22 ข้อจำกัดและ non-goals 13

23 สรุป 13

1 สถานะของเอกสารและขอบเขตหลักฐาน

ตาราง 1 แยกสิ่งที่มีหลักฐานใน repository ออกจากวิธีที่กำลังเสนอ เพื่อป้องกัน การนำ architecture ไปกล่าวเหมือนเป็นผลทดลองแล้ว

ตารางที่ 1: สถานะขององค์ประกอบที่กล่าวถึงในรายงาน

องค์ประกอบ	สถานะ	ความหมาย
AGSI++ ราย IBR	VERIFIED CURRENT	มีสมการ bounded index, threshold, hysteresis และ dwell ใน production code
GFL/GFM full-state และ full KCL	VERIFIED CURRENT	มี route การจำลองและ transaction ที่ตรวจด้วย targeted tests
SG-trip override/SG handback	VERIFIED CURRENT	มี chronology ที่สลับ reference และ mode แบบ coordinated
ET-FCSPS system decision	PROJECT-DERIVED PROPOSAL	architecture, equations และ test plan ในรายงานนี้
ผลว่าลด transient/จำนวน switch	NOT YET IMPLEMENTED	ต้อง implement และเปรียบเทียบกับ baseline บน contract เดียวกัน
AVR, LLM/BO online switching	นอกขอบเขต	ไม่ถูกเพิ่มหรือใช้เป็น decision authority

คำว่า “adaptive” ในรายงานนี้หมายถึงการเลือก mode vector ตาม accepted system state, topology และ reserve ณ เหตุการณ์นั้น ไม่ได้หมายถึง online parameter identification หรือการปรับ weight โดยอัตโนมัติ ส่วนคำว่า finite-control-set ใช้หลักการประเมิน action set แบบจำกัดที่แจกแจงได้ โดยประยุกต์จากแนวคิด FCS-MPC [1]; ไม่ได้หมายถึงการสั่ง semiconductor switch ระดับ converter โดยตรง

2 ปัญหาของ AGSI++ แบบ local-only

2.1 สมการที่ใช้อยู่ปัจจุบัน

สำหรับ IBR ตัวที่ i production class คำนวณ sub-index ที่ PCC ดังนี้

$$J_{V,i} = \frac{||V_i| - V_{ref,i}|}{\Delta V_{base}}, \qquad J_{f,i} = \frac{|f_i - f_0|}{\Delta f_{base}}, \tag{1}$$

$$J_{R,i} = \frac{|\dot{f}_i|}{\Delta R_{base}}, \qquad J_{P,i} = \frac{|P_{ref,i} - P_i|}{\Delta P_{base}}, \tag{2}$$

$$J_{SCR,i} = \max\left(0, \frac{3}{SCR_i} - 1\right), \qquad J_{lock,i} = \frac{|v_{q,i}|}{\Delta v_{q,base}}, \tag{3}$$

$$J_{GRA,i} = \begin{cases} 0, & GRA_i = 1, \\ 1, & GRA_i = 0. \end{cases} \tag{4}$$

ดัชนีรวมคือ

$$A_i = \mathrm{sat}_{[0,1]}\left(w_V J_{V,i} + w_f J_{f,i} + w_R J_{R,i} + w_P J_{P,i} + w_{SCR} J_{SCR,i} + w_{lock} J_{lock,i} + w_{GRA} J_{GRA,i}\right), \tag{5}$$

โดย $\sum w = 1$ และ AGSI++ preset ใช้ค่าเฉลี่ยเท่ากัน $1/7$ สำหรับเจ็ดพจน์ RoCoF ผ่าน first-order low-pass ก่อนเข้าดัชนี ส่วน raw weighted sum ถูกเก็บเพื่อวิเคราะห์แต่ decision index ถูกจำกัดให้อยู่ใน $[0, 1]$ เสมอ

logic การสลับ local คือ

$$\text{GFL} \rightarrow \text{GFM}: \quad A_i \geq \Gamma_{on} \text{ ต่อเนื่องอย่างน้อย } T_{d,on}, \tag{6}$$

$$\text{GFM} \rightarrow \text{GFL}: \quad A_i < \Gamma_{off} \text{ ต่อเนื่องอย่างน้อย } T_{d,off}, \tag{7}$$

โดย $\Gamma_{off} < \Gamma_{on}$ สร้าง hysteresis band ป้องกัน chatter ค่าใน report route ปัจจุบันคือ $\Gamma_{on} = 0.65$, $\Gamma_{off} = 0.35$, $T_{d,on} = 0.10$ s และ $T_{d,off} = 1.00$ s ค่าดังกล่าวจะถูก freeze เป็น baseline; รายงานนี้ไม่อนุญาตให้เปลี่ยนเพื่อทำให้ผลของวิธีใหม่ดีขึ้น

2.2 AGSI++ มองเห็นอะไรและยังไม่เห็นอะไร

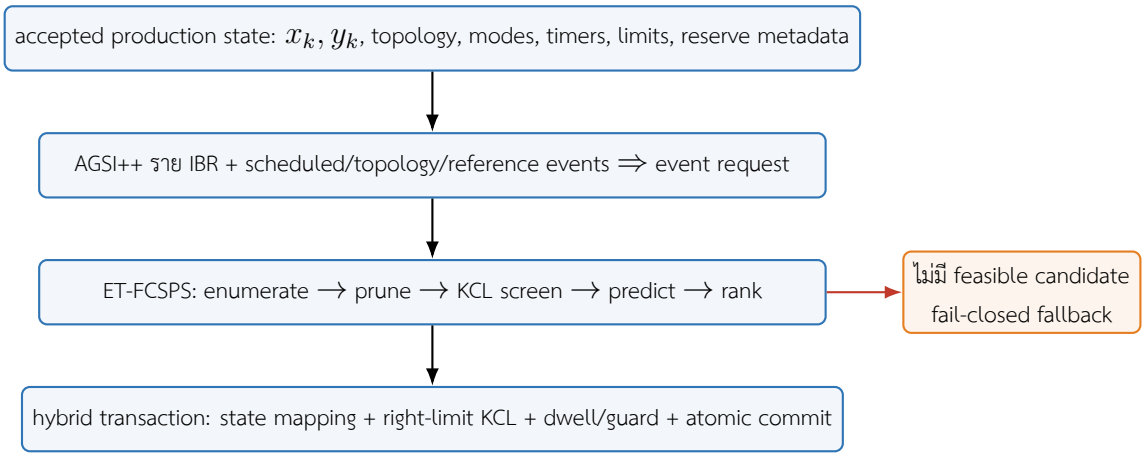
AGSI++ ไม่ใช่ข้อมูลน้อย: มันมี voltage, frequency, RoCoF, active-power tracking, SCR, PLL alignment และ reference availability แต่ทุกพจน์ผูกกับ IBR ตัวเดียวหรือ PCC ของตัวนั้น จึงยังไม่มีการประเมินพร้อมกันว่า

- voltage violation เกิดที่บัสอื่นหรือมีขอบเขตกว้างเพียงใด;
- topology ปัจจุบันทำให้ IBR ตัวใดอยู่ใกล้พื้นที่อ่อนในความหมายทางไฟฟ้า;
- active/reactive reserve ของชุด IBR ที่จะเปลี่ยนโหมดเพียงพอหรือไม่;
- การสลับหนึ่งตัวหรือหลายตัวจะเพิ่ม current-limit exposure ที่จุดอื่นหรือไม่;
- energized island ทุกเกาะมี reference owner และ owner ที่เลือกมี headroom หรือไม่;
- candidate หนึ่งจะให้ transient หลัง commit ดีกว่า candidate อื่นหรือไม่

ดังนั้นข้อจำกัดไม่ใช่ “AGSI ไม่มีแรงดันหรือกำลัง” แต่คือมันยังไม่ได้ใช้ full-network consequence เป็น decision authority ก่อน commit

3 แนวคิดหลักของ ET-FCSPS

ET-FCSPS วางชั้นตัดสินใจระดับระบบเหนือ local state machine โดยไม่ยกเลิก AGSI++ โครงสร้างเป็นไปตามรูป 1



รูปที่ 1: สถาปัตยกรรมที่เสนอ: AGSI++ เป็น local evidence ส่วน ET-FCSPS เป็นผู้อนุมัติคำสั่งระดับระบบ

หลักสำคัญมีสี่ข้อ

1. **Event-triggered**: ไม่รัน candidate search ทุก integration step แต่รันเมื่อมี local request, topology/reference event หรือ post-event re-evaluation ตาม contract
2. **Finite action set**: แจกแจง mode vector ที่เป็นไปได้ทั้งหมด จึง audit ได้ว่าได้ตรวจตัวเลือกใด
3. **Constraint first**: candidate ที่ผิด KCL, limit, reserve หรือ reference gate ถูกตัดก่อนดู cost
4. **Predict before commit**: candidate trial ทำงานบนสำเนา state; state จริงเปลี่ยนเฉพาะ atomic commit

4 นิยามข้อมูลเข้าและ system snapshot

เมื่อ controller ถูกเรียกที่ accepted sample k ให้สร้าง snapshot

$$\mathcal{Z}_k = \{t_k, x_k, y_k, \mathcal{T}_k, \mathsf{m}_k, \boldsymbol{\tau}_k, \mathcal{L}_k, \mathcal{R}_k, \mathcal{E}_k\}, \tag{8}$$

โดยตัวแปรมีความหมายตามตาราง 2

ตารางที่ 2: ข้อมูลใน accepted system snapshot

สัญลักษณ์	ความหมาย
t_k	เวลาของ accepted sample; ห้ามใช้ rejected parent trial
x_k	differential states ของ SG และ IBR ตาม production state order
y_k	algebraic network voltages/currents ของ full-KCL solve
$\mathcal{T}_k$	topology, energized islands, line/breaker status และ event side
m_k	committed mode vector ของ IBR ณ ปัจจุบัน
$\boldsymbol{\tau}_k$	up/down dwell timers, lockout และ last-switch time ราย IBR
$\mathcal{L}_k$	voltage/current/operating limits ที่มี provenance
$\mathcal{R}_k$	active/reactive reserve และ current headroom; ถ้าไม่ทราบต้องเป็น unknown ไม่ใช่ ∞
$\mathcal{E}_k$	event identity, AGSI parts, GRA และ reference owner ปัจจุบัน

global metrics ที่ controller ใช้ต้องคำนวณจาก base และ sign convention เดียวกับ KCL เช่น

$$V_{min,k} = \min_b |V_{b,k}|, \qquad V_{max,k} = \max_b |V_{b,k}|, \tag{9}$$

$$\Delta P_{avail,k} = P_{SG}^{avail} + \sum_i P_i^{avail} - P_{load} - P_{loss}, \tag{10}$$

$$\Delta Q_{avail,k} = Q_{SG}^{avail} + \sum_i Q_i^{avail} - Q_{load} - Q_{loss}. \tag{11}$$

ต้องไม่ใช้ $\sum P_{electrical} - P_{load} - P_{loss}$ จาก algebraic solution เป็น reserve metric เพราะ KCL ทำให้ผลรวมนั้นใกล้ศูนย์โดยนิยาม สิ่งที่ต้องใช้คือ available/commandable source power และ headroom ซึ่งต้องมี source/case contract หาก energy-source model ยังไม่ระบุ candidate ที่ต้องพึ่ง reserve ดังกล่าว ต้องถูกจัดเป็น “ไม่ทราบ” และไม่อาจผ่าน production gate ด้วยการสมมติค่าเอง

5 decision variables และ candidate universe

สำหรับ IBR สี่ตัว กำหนด

$$\mathsf{m} = [m_1, m_2, m_3, m_4] \in \{0,1\}^4, \qquad m_i = 0 : \text{GFL}, \quad m_i = 1 : \text{GFM}. \tag{12}$$

จึงมี mode vector $2^4 = 16$ รูปแบบ ถ้า SG online และเป็น reference owner คงที่ candidate universe มีไม่เกิน 16 รูปแบบ ถ้า SG offline และยอมให้ GFM ทุกตัวในชุดเป็น owner candidate ได้ จำนวนคู่ (m, r) สูงสุดคือ

$$\sum_{\emptyset \neq S \subseteq \{1,2,3,4\}} |S| = 4\,2^{4-1} = 32. \tag{13}$$

นี่เป็นขนาดเล็กพอให้แจกแจงครบโดยไม่ต้องใช้ black-box optimiser อย่างไรก็ตาม candidate ที่มี mode เดิม, ติด dwell/lockout, ไม่มี owner หรือไม่มี state mapping ที่ valid จะถูก prune ก่อน prediction

reference variable นิยามเป็น

$$r \in \{\text{SG}, 1, 2, 3, 4\}. \tag{14}$$

ถ้า $r = i$ ต้องมี $m_i = 1$ และ energized island นั้นต้องมี owner เพียงหนึ่งตัว คำว่า owner หมายถึง ผู้ให้ angle/frequency reference ใน formulation ของ island ไม่ใช่การเปลี่ยน IBR ให้เป็น algebraic PF slack; ทุกอุปกรณ์ยังอยู่ใน full-network KCL

6 event trigger และ candidate generation

supervisor ถูกเรียกเมื่ออย่างน้อยหนึ่งเงื่อนไขเป็นจริง

1. local state machine รายงาน pending GFL⟶GFM หรือ GFM⟶GFL request;
2. SG/breaker/reference availability เปลี่ยน;
3. topology หรือ load event ถูก apply/restore;
4. fault apply/clear;
5. synchronism guard ครบ dwell และต้องประเมิน handback;
6. post-commit verification interval ถึงเวลาตาม contract

ไม่มีการสร้าง threshold ใหม่เพื่อเรียก controller หากยังไม่มี source/approved derivation; ในเฟสแรกให้ reuse event identities และ local requests ที่มีอยู่

candidate generation ทำตามลำดับ

1. ระบุ energized islands จาก $\mathcal{T}_k$
2. แจกแจง m และ owner r
3. ตัด candidate ที่ขัดกับ device capability, dwell, lockout และ reference rule
4. map state บนสำเนา x_k ตาม GFL/GFM transfer contract
5. สร้าง candidate fingerprint จาก snapshot identity, mode vector, owner และ option contract

7 hard feasibility gates

candidate $c = (\mathsf{m}, r)$ ต้องผ่านทุก gate ก่อนคำนวณ soft cost

$$g_{finite}(c): \quad x^{(c)}, y^{(c)} \text{ finite}, \tag{15}$$

$$g_{KCL}(c): \quad \|F(x^{(c)}, y^{(c)}, c)\|_\infty \le \varepsilon_{KCL}, \tag{16}$$

$$g_V(c): \quad V_{min}^{lim} \le |V_b^{(c)}| \le V_{max}^{lim} \quad \forall b, \tag{17}$$

$$g_I(c): \quad |I_i^{(c)}| \le I_{i,max} \quad \forall i, \tag{18}$$

$$g_{ref}(c): \quad \text{ทุก energized island มี authenticated owner หนึ่งตัว}, \tag{19}$$

$$g_{reserve}(c): \quad \Delta P_{avail}^{(c)} \ge 0, \quad \Delta Q_{avail}^{(c)} \ge 0, \tag{20}$$

$$g_{hyb}(c): \quad \text{state mapping, dwell, lockout และ transaction identity valid}. \tag{21}$$

ค่าขอบเขตและ tolerance ต้องมาจาก production/case contract และถูก freeze ก่อนเห็นผล comparison ห้ามเลือก candidate ที่ผิด hard constraint เพียงเพราะ cost ต่ำกว่า

7.1 algebraic screening

ชั้นแรกใช้ project-owned PF/full-KCL solver ประเมิน right-limit ของ candidate โดยไม่ advance เวลา ผลลัพธ์ต้องเก็บ converged flag, residual, voltage range, current margin, reserve status และ rejection reason candidate ที่ Newton ไม่ converge ต้องถูก reject ไม่ใช่แทนด้วยค่าจาก candidate ก่อนหน้า

7.2 สิ่งที่ยังไม่ใช่ feasibility proof

PF convergence อย่างเดียวไม่พิสูจน์ transient stability และ local eigenvalue อย่างเดียวไม่พิสูจน์ผลของ multi-IBR island ดังนั้น candidate ที่ผ่าน algebraic gate ยังต้องผ่าน short-horizon prediction และ runtime domain gates การที่ objective ต่ำก็ไม่ใช่ข้อยกเว้นให้ข้ามขั้นเหล่านี้

8 short-horizon prediction

สำหรับ candidate ที่ผ่าน algebraic screen ให้สร้าง isolated trial context

$$(\hat{x}_0^{(c)}, \hat{y}_0^{(c)}) = \mathcal{M}_c(x_k, y_k), \tag{22}$$

โดย $\mathcal{M}_c$ คือ audited mode-state mapping แล้วจำลอง

$$\dot{\hat{x}}^{(c)} = f(\hat{x}^{(c)}, \hat{y}^{(c)}, c, u_k), \tag{23}$$

$$0 = F(\hat{x}^{(c)}, \hat{y}^{(c)}, c, \mathcal{T}_k), \qquad t \in [t_k, t_k + T_p]. \tag{24}$$

T_p เป็น prediction horizon ที่ต้องประกาศก่อน experiment ในเฟสแรกให้ทดสอบหลายค่าที่ predeclared เพื่อรายงาน sensitivity ไม่เลือกค่าหลังเห็นว่า candidate ใดชนะ

กฎ isolation คือ

- trial ทุกตัวเริ่มจาก accepted snapshot เดียวกัน;
- ห้ามแก้ production device object, timer, RoCoF memory หรือ event log ระหว่าง trial;
- trial ที่ออกจาก validity domain, residual gate หรือ limit ต้องหยุดและบันทึก fail reason;
- หลังประเมินแล้วทิ้ง trial state ทั้งหมด; commit ได้เพียง winner หลังตรวจ fingerprint ซ้ำ

9 objective function และ normalization

ให้ $\mathcal{C}_k^{feas}$ เป็นชุด candidate ที่ผ่าน hard gates เลือก

$$c_k^* = \arg\min_{c \in \mathcal{C}_k^{feas}} J(c; \mathcal{Z}_k), \tag{25}$$

โดยตัวอย่าง objective ที่เสนอคือ

$$\begin{aligned} J(c) = {} & w_V J_V(c) + w_f J_f(c) + w_R J_{RoCoF}(c) + w_P J_{P,res}(c) + w_Q J_{Q,res}(c) \\ & + w_I J_I(c) + w_H J_{hand}(c) + w_N J_{GFM}(c) + w_S J_{switch}(c). \end{aligned} \tag{26}$$

พจน์ทั้งหมดต้อง dimensionless ก่อนถ่วงน้ำหนัก ตัวอย่างนิยามคือ

$$J_V(c) = \max_{t,b} \left[\frac{(|V_b| - V_{max}^{lim})_+ + (V_{min}^{lim} - |V_b|)_+}{\Delta V_{norm}} \right], \tag{27}$$

$$J_f(c) = \max_t \frac{|f_{ref}(t) - f_0|}{\Delta f_{norm}}, \tag{28}$$

$$J_{RoCoF}(c) = \max_t \frac{|\dot{f}_{ref}(t)|}{\Delta R_{norm}}, \tag{29}$$

$$J_I(c) = \max_{t,i} \frac{|I_i(t)|}{I_{i,max}}, \tag{30}$$

$$J_{GFM}(c) = \frac{1}{4} \sum_i m_i, \tag{31}$$

$$J_{switch}(c) = \frac{1}{4} \sum_i |m_i - m_{i,k}|, \tag{32}$$

$$J_{hand}(c) = \max \left(\frac{|\Delta P|}{\Delta P_{norm}}, \frac{|\Delta Q|}{\Delta Q_{norm}}, \frac{|\Delta I|}{\Delta I_{norm}}, \frac{|\Delta \theta|}{\Delta \theta_{norm}} \right). \tag{33}$$

เมื่อ $(a)_+ = \max(a, 0)$ ส่วน $J_{P,res}$ และ $J_{Q,res}$ วัดระยะใกล้ขอบ reserve ไม่ใช่ electrical KCL mismatch

9.1 กติกาการกำหนด weight

weight ทุกตัวต้องไม่ติดลบและ $\sum w = 1$ การกำหนดต้องจัดประเภท SOURCE_DEFINED, CASE_DEFINED หรือ PROJECT_DERIVED และ freeze ก่อนผล comparison หากยังไม่มีเหตุผลรองรับ ให้รายงาน sensitivity grid ที่กำหนดล่วงหน้าแทนการหา weight ที่ทำให้ผลสวยที่สุด BO อาจใช้เป็น offline comparison ได้ในอนาคต แต่ไม่ใช่ส่วนจำเป็นของ ET-FCSPS และ output ของ BO ห้าม feed production ก่อนผ่าน provenance และ validation contract

10 deterministic ranking และ tie-break

เพื่อให้ทำซ้ำได้ candidate ordering ใช้ลำดับต่อไปนี้

1. ผ่าน hard gates ทุกข้อก่อน candidate ที่ไม่ผ่าน;
2. $J(c)$ ต่ำกว่า;
3. จำนวน switching actions ต่ำกว่า;
4. จำนวน GFM ต่ำกว่า;
5. reserve margin ต่ำสุดของ candidate สูงกว่า;
6. accepted residual ต่ำกว่า;
7. lexicographic mode vector และ owner ID เป็น tie-break สุดท้าย

ต้องกำหนด floating-point comparison tolerance ล่วงหน้าและเก็บ ordering key ของทุก candidate ห้ามพึ่ง iteration order ของ cell/struct หรือ random seed

11 reference leader และ transaction semantics

11.1 SG trip หรือ reference loss

เมื่อ SG เปิดและ $GRA = 0$ controller ต้องเลือก (m, r) ที่มีอย่างน้อยหนึ่ง GFM จากนั้น commit เป็น transaction เดียว

1. freeze accepted left-limit state;
2. authenticate event และ candidate fingerprint;
3. map ทุก branch ตาม mode vector;
4. กำหนด owner r และ reference frame;
5. solve right-limit KCL;
6. ถ้าผ่านทุก gate จึงเปลี่ยน committed state/mode/timer พร้อมกัน;
7. ถ้าไม่ผ่าน rollback ทั้ง transaction

11.2 SG reclose และ handback

การคืน reference ต้องตรวจ synchronism guard

$$|\Delta V| \leq \Delta V_{max}, \qquad |\Delta f| \leq \Delta f_{max}, \qquad |\Delta \theta| \leq \Delta \theta_{max} \tag{34}$$

ครบ dwell ก่อน close จากนั้น controller ประเมิน candidate ที่ SG เป็น owner ซึ่งอาจเป็น all-GFL หรือคงบาง IBR เป็น GFM ชั่วคราว หาก policy อนุญาต การ handback ต้องใช้ state mapping และ right-limit KCL เดียวกับ forward switch เพื่อลด jump และป้องกัน partial commit

12 fail-closed fallback

ถ้า $\mathcal{C}_k^{feas} = \emptyset$ ห้ามเลือก candidate ที่ผิด constraint แผน fallback เฟสแรกคือ

1. ถ้า current committed configuration ยังผ่าน hold gate ให้คง mode เดิมและบันทึก no-decision;
2. เมื่อเกิด authenticated SG/reference loss ให้ลอง current verified atomic all-GFM emergency route เฉพาะเมื่อ route นั้นผ่าน right-limit gates;
3. ถ้า emergency route ไม่ผ่าน ให้ simulation/controller return explicit infeasible status;
4. ห้ามเติม trajectory, freeze graph หรือแทน state ด้วย candidate อื่น;
5. protection action ของระบบจริงอยู่นอก model และต้องออกแบบแยกต่างหาก

13 pseudocode ของ controller

```
function decision = et_fcs_supervisor(accepted_snapshot, event)
assert(snapshot is accepted, finite, fingerprinted)
if not local_request and not authenticated_event: return HOLD
C = enumerate_mode_owner_pairs(snapshot)
C = prune_by_capability_dwell_lockout_reference(C)
feasible = []
for each c in C in canonical order
trial = deep_copy(snapshot)
trial = map_states_for_candidate(trial,c)
a = solve_right_limit_full_KCL(trial,c)
if not algebraic_gates_pass(a): log reject; continue
p = simulate_short_horizon_isolated(a,c,Tp)
if not dynamic_gates_pass(p): log reject; continue
feasible.append(c, metrics(p), cost(p), ordering_key)
end
if feasible is empty: return declared_fail_closed_fallback()
winner = canonical_argmin(feasible)
revalidate(snapshot_fingerprint, event, winner)
return atomic_commit_request(winner, evidence_table)
end
```

รูปที่ 2: pseudocode ที่เสนอ; ชื่อฟังก์ชันเป็น interface plan ยังไม่ใช่ไฟล์ production

14 computational complexity และการลด runtime

ถ้าประเมินครบ N_c candidates และแต่ละ prediction ใช้ $N_p = T_p/\Delta t$ steps ต้นทุนโดยคร่าวคือ

$$\mathcal{O}(N_c\,[C_{KCL} + N_p C_{step}])\,, \tag{35}$$

โดย $N_c \leq 16$ เมื่อ SG เป็น owner และ $N_c \leq 32$ เมื่อเลือก IBR owner ด้วย วิธีลดเวลาโดยไม่ลด numerical integrity คือ

1. logical pruning ก่อนเรียก solver;
2. algebraic screen ก่อน dynamic prediction;
3. หยุด candidate ทันทีเมื่อผิด hard gate;
4. ทำนายเฉพาะ event-trigger ไม่ใช่ทุก time step;

5. parallel candidate evaluation ได้เมื่อ trial object แยกกันจริงและผล sorting deterministic;
6. cache ได้เฉพาะ immutable equilibrium ที่ fingerprint ตรงทุก input

production chronology ปัจจุบันใช้เวลาระดับหลายสิบนาที จึงยังห้ามอ้าง real-time capability ก่อนมี profiling ของ controller deadline, worst-case candidate count และ solver iteration distribution

15 ผลลัพธ์ที่คาดหวังและวิธีพิสูจน์

ผลที่คาดหวังในตาราง 3 เป็น hypotheses ไม่ใช่ผลทดลอง

ตารางที่ 3: ประโยชน์ที่คาดหวัง กลไก และ metric

ผลที่คาดหวัง	กลไก	metric
ลดการสลับที่ไม่จำเป็น	objective คำนึงถึงจำนวน GFM และจำนวนครั้งที่เปลี่ยน mode	switch count, chatter, GFM-on time
ลด voltage/ frequency transient	ทำนายก่อน commit และตัด candidate ที่ excursion สูง	peak $\lvert\Delta f\rvert$, RoCoF, voltage violation, settling time
ลด current/limiter stress	current/reserve เป็น hard gate และ cost margin	max $\lvert I\rvert/I_{max}$, limiter dwell, reserve margin
ทำ handback ให้นุ่มขึ้น	เปรียบเทียบ jump ของ candidate ก่อน atomic commit	$\Delta P, \Delta Q, \Delta I, \Delta\theta$ ที่ event
รักษา reference continuity	บังคับทุก energized island มี owner	no-reference duration ต้องเป็นศูนย์
อธิบายย้อนหลังได้	เก็บ candidate evidence และ deterministic ordering	decision table reproduce ได้จาก snapshot

การพิสูจน์ต้องใช้ baseline AGSI-only และ ET-FCSPS บน case, event, parameter, Δt, solver options และ gates เดียวกัน รายงานทั้ง improvement และ regression ห้ามเลือกเฉพาะเหตุการณ์ที่ method ใหม่ชนะ

16 การเปรียบเทียบกับ Bayesian optimisation

BO สร้าง probabilistic surrogate ของ expensive black-box objective แล้วใช้ acquisition function เลือกจุดทดลอง ถัดไป [3, 4] จึงเหมาะกับ offline parameter/hyperparameter search ที่ evaluation แพง ขณะที่ constrained MPC ใช้ current state เป็น initial condition แก้ finite-horizon problem และใช้ action แรก โดยมีจุดเด่นด้าน state/control constraints [2]

ตารางที่ 4: บทบาทของ BO และ ET-FCSPS ในโจทย์นี้

ประเด็น	BO	ET-FCSPS
โจทย์หลัก	ค้น parameter ของ expensive black-box objective	เลือก action จาก finite mode/owner set ณ accepted state
ข้อมูล	evaluations เพื่อสร้าง surrogate/uncertainty	project plant model + full-network snapshot
constraint	ต้องออกแบบ constrained BO เพิ่ม	reject hard-constraint violation ก่อน ranking
traceability	ต้องอธิบาย surrogate, kernel และ acquisition	แสดง gate/metric/cost ของทุก candidate
ต้นทุนหลัก	จำนวน expensive samples	จำนวน candidate คูณ prediction horizon
บทบาทที่เหมาะ	offline design/tuning study	event-time supervisory action selection

จึงควรอธิบายว่า ET-FCSPS “เหมาะกับ decision set เล็กและต้อง audit ได้ในงานนี้” ไม่ใช่ “ดีกว่า BO ทุกกรณี” หากงานรุ่นก่อนใช้ BO เพื่อปรับ weight offline ทั้งสองวิธีกำลังทำคนละหน้าที่ และ comparison ที่เป็นธรรม ต้องใช้ objective/data budget/event set เดียวกัน

17 แผน interface สำหรับ implementation

ชื่อไฟล์ในตาราง 5 เป็นข้อเสนอ ยังไม่มี production implementation

ตารางที่ 5: interface ที่เสนอและหน้าที่

ชื่อที่เสนอ	หน้าที่
`+stability/et_fcs_snapshot.m`	สร้าง immutable accepted snapshot และ fingerprint
`+stability/et_fcs_enumerate.m`	แจกแจง mode/owner pairs และ canonical IDs
`+stability/et_fcs_screen.m`	state mapping, right-limit KCL และ hard gates
`+stability/et_fcs_predict.m`	isolated short-horizon trial
`+stability/et_fcs_metrics.m`	global voltage/frequency/current/reserve metrics
`+stability/et_fcs_rank.m`	normalization, cost และ deterministic tie-break
`+stability/et_fcs_supervisor.m`	orchestration และ fail-closed decision
`tests/test_et_fcs_*.m`	unit/oracle/integration/falsification tests

output schema ขั้นต่ำต้องมี event ID, snapshot fingerprint, candidate ID, mode vector, owner, prune/reject reason, right-limit residual, predicted metrics, normalized costs, ordering key, winner identity, runtime และ commit/rollback result

18 แผนพัฒนาเป็นเฟส

1. **Freeze contract**: state order, base/sign, limits, reserve semantics, event identity และ baseline outputs
2. **Global metrics**: implement voltage/topology/reference/current metrics พร้อม independent identities
3. **Enumeration oracle**: พิสูจน์ 16 mode vectors และ 32 mode-owner pairs ไม่ซ้ำและ ordering deterministic
4. **Algebraic screening**: clone/mapping/right-limit KCL โดยไม่มี side effect
5. **Prediction and cost**: short-horizon isolation, metric normalization และ predeclared weight set
6. **Transaction integration**: atomic commit, fingerprint revalidation, rollback และ fallback
7. **Falsification**: no-event, load, fault, line trip, SG trip/reclose, dt/horizon/runtime sensitivity
8. **Baseline comparison**: AGSI-only vs ET-FCSPS บน identical contracts

19 verification matrix

ตารางที่ 6: เกณฑ์ตรวจที่ต้องประกาศก่อนผล

Gate	สิ่งที่ตรวจ	หลักฐาน
Enumeration	mode/owner universe ครบ ไม่ซ้ำ	independent combinatorial oracle
Snapshot isolation	trial ไม่เปลี่ยน production object/state/timer	before/after fingerprint bit identity
Metric identity	$P, Q, loss, current, reserve$ ใช้ base/sign ถูกต้อง	direct KCL/power identities
Feasibility	candidate ผิด constraint ถูก reject	injected counterexamples ทุก gate
Ranking	winner เป็น canonical minimum	independent sorted table oracle
Hybrid transaction	mapping/right-limit/rollback atomic	no partial commit และ finite residual

Gate	สิ่งที่ตรวจ	หลักฐาน
Reference security	ทุก energized island มี owner	zero no-reference accepted duration
Behaviour	no-event/load/fault/line/SG events	ไม่มี false switch/chatter และ log มีเหตุผล
Numerical	$\Delta t, T_p$, finite-difference sensitivity	รายงานความต่างโดยไม่ tune tolerance
Runtime	candidate count, solver iterations, deadline	percentile/ worst-case พร้อม timeout fall-back
Comparison	baseline และ proposal ใช้ contract เดียวกัน	paired scenario table ไม่คัดเฉพาะผลดี

full repository regression ไม่จำเป็นทุก edit ระหว่างพัฒนา แต่ targeted gates ต้องครอบคลุม producer, consumer และ failure path ทุกเฟส หลัง integration ที่เปลี่ยน shared runtime ให้ใช้ risk policy ของ repository กำหนดว่าต้องขยาย regression หรือไม่

20 ตัวอย่างลำดับการตัดสินใจ

20.1 SG trip

ที่ event SG trip, GRA เปลี่ยนเป็นศูนย์และ controller ถูก trigger สมมติ logical pruning เหลือ candidate ที่มี GFM อย่างน้อยหนึ่งตัวและ owner อยู่ใน GFM set จากนั้น KCL screen อาจตัดชุดที่ voltage/current gate ไม่ผ่าน prediction เปรียบเทียบชุดที่เหลือ หาก $(1000, r = 1)$ ผ่านและ cost ต่ำที่สุด controller จึง commit IBR1 เป็น GFM owner ส่วน IBR2–4 คง GFL ไม่ใช่สั่ง all-GFM อัตโนมัติ แต่ถ้าชุดหนึ่งตัวไม่ผ่านและ $(1111, r = 1)$ เป็นชุดเดียวที่ feasible การเลือก all-GFM ก็มีเหตุผลจาก evidence table ไม่ใช่เพราะ AGSI ชนพร้อมกัน

20.2 fault เฉพาะจุด

AGSI ของ IBR ใกล้ fault อาจ trigger ก่อนตัวอื่น แต่ supervisor ยังประเมิน bus-voltage violation, electrical position, current headroom และ predicted transient ของทุก admissible set ผลอาจเลือก IBR ใกล้ fault, เลือก IBR ที่มี reserve มากกว่า หรือไม่สลับเลยถ้า ride-through configuration ผ่านทุก gate ทั้งหมดเป็น hypothesis จนกว่าจะรัน predictor จริง

20.3 SG reclose

เมื่อ synchronism guard ผ่าน supervisor สร้าง candidate ที่ SG กลับเป็น owner เปรียบเทียบ all-GFL กับชุดที่ คง GFM ชั่วคราวโดยพิจารณา handback jump และ transient ถ้า all-GFL ผ่านและมี cost ต่ำจึง commit คืนโหมด หากไม่มี candidate ผ่านต้อง hold/fail-closed ไม่บังคับกลับเพียงเพราะ $A_i < \Gamma_{off}$

21 แนวตอบคำถามทางเทคนิค

ถาม: ยังใช้ AGSI++ อยู่หรือไม่?

ใช้ แต่เปลี่ยนบทบาทจากผู้สั่งโดยตรงเป็น local event indicator และ evidence ราย IBR การอนุมัติคำสั่งอยู่ที่ system-level supervisor

ถาม: adaptive ตรงไหนถ้า weight ไม่เปลี่ยน?

คำสั่ง mode/owner ปรับตาม accepted network state, topology, limits และ predicted consequences ทุก event จึงเป็น state-adaptive action selection ไม่ใช่ adaptive parameter identification

ถาม: ทำไมไม่ใช้ LLM?

คำสั่งต้อง deterministic, bounded, replayable และ fail-closed LLM ไม่มี numerical/constraint guarantee จึงเหมาะกับการอธิบาย log แบบ offline ไม่ใช่ decision authority

ถาม: ทำไมไม่ใช้ BO เหมือนงานเดิม?

BO เหมาะกับการหา parameter ของ objective ที่ประเมินแพงแบบ offline ส่วนงานนี้มี discrete action set เพียง 16 mode vectors และต้อง enforce hard constraints ณ event จึงแจกแจงครบและ audit ได้ตรงกว่า

ถาม: วิธีนี้รับประกันลด transient หรือไม่?

ยังไม่รับประกัน เป็น hypothesis ที่ต้องเทียบ peak frequency/RoCoF, voltage violation, current stress, settling time และ handback jump กับ AGSI-only baseline บน input contract เดียวกัน

ถาม: IBR1 กลายเป็น slack bus หรือไม่?

ไม่ IBR1 อาจเป็น reference owner ใน island แต่ยังเป็น dynamic current-injection device ใน full KCL PF slack เป็นคนละความหมาย

ถาม: ถ้าไม่มี candidate ผ่านจะทำอย่างไร?

ใช้ declared fail-closed fallback: hold current safe configuration หรือ verified emergency transaction ถ้าผ่าน right-limit gates มิฉะนั้นคืน infeasible status ไม่ปลอม trajectory

ถาม: พร้อม real-time หรือยัง?

ยัง ต้องมี runtime profiling และ deadline gate เพราะ short-horizon full-KCL prediction มีต้นทุนสูง

22 ข้อจำกัดและ non-goals

1. ET-FCSPS ยังไม่ถูก implement และไม่มี closed-loop performance result
2. ไม่เพิ่ม AVR; E_{fd} และ voltage control contract ของ SG คงตาม model ปัจจุบัน
3. ไม่แก้สมการ SG/IBR, AGSI++, threshold, case data หรือ event chronology ในขั้นรายงาน
4. reserve/energy availability ต้องมี source/case definition ก่อนใช้เป็น production gate
5. short horizon ไม่รับประกัน infinite-horizon stability โดยตัวมันเอง; ต้องมี falsification และ stability analysis เพิ่ม
6. model mismatch, communication delay, protection hardware และ cyber-security อยู่นอกผลจำลองปัจจุบัน
7. วิธีนี้ไม่ถูกกล่าวว่าดีกว่า BO ทุกกรณี และไม่ใช้ BO/LLM เป็น online switching authority

23 สรุป

ET-FCSPS แก้ช่องว่างของ AGSI-only switching โดยแยก “การตรวจพบความเครียดเฉพาะจุด” ออกจาก “การอนุมัติคำสั่งระดับระบบ” AGSI++ ยังคงตรวจ local condition และ dwell ขณะที่ supervisor แจกแจง mode/owner set, ตัด candidate ที่ผิด hard constraints, ทำนายผลช่วงสั้น แล้วเลือกคำสั่งที่ทำซ้ำและอธิบายได้ สำหรับสี่ IBR ขนาด candidate สูงสุด 16 mode vectors หรือ 32 mode-owner pairs ทำให้ exhaustive evaluation เป็นทางเลือกที่ตรวจสอบได้โดยไม่ต้องใช้ black-box online optimiser

ประโยชน์ที่คาดหวังคือใช้ GFM เท่าที่จำเป็น ลด switching/transient/current stress ทำ handback นุ่มขึ้น รักษา reference continuity และสร้าง evidence table ของทุกการตัดสินใจ แต่ทั้งหมดต้องพิสูจน์ด้วย implementation และ paired comparison บน numerical contract เดียวกันก่อนจึงจะกล่าวเป็นผลของงานได้

References

[1] J. Rodriguez, M. P. Kazmierkowski, J. R. Espinoza, P. Zanchetta, H. Abu-Rub, H. A. Young, and C. A. Rojas, “State of the Art of Finite Control Set Model Predictive Control in Power Electronics,” *IEEE Transactions on Industrial Informatics*, vol. 9, no. 2, pp. 1003–1016, 2013. DOI: 10.1109/TII.2012.2221469.

[2] D. Q. Mayne, J. B. Rawlings, C. V. Rao, and P. O. M. Scokaert, “Constrained Model Predictive Control: Stability and Optimality,” *Automatica*, vol. 36, no. 6, pp. 789–814, 2000. DOI: 10.1016/S0005-1098(99)00214-9.

[3] P. I. Frazier, “A Tutorial on Bayesian Optimization,” arXiv:1807.02811, 2018.

[4] J. Snoek, H. Larochelle, and R. P. Adams, “Practical Bayesian Optimization of Machine Learning Algorithms,” *Advances in Neural Information Processing Systems 25*, 2012.

[5] Power-flow repository: `+ibr/SwitchableIbr6.m`, `+stability/ibr_selector_table.m`, and the audited full-KCL hybrid runtime. Accessed 4 August 2026.