

Алгоритми управління насосними групами і станціями.

В.О.Дмитренко,
Г.А.Сікорський,
Ведучі спеціалісти НТТОВ ЕЛЕКОН.

Групи насосів найрізноманітнішого призначення і насосні станції є одним з найбільш поширених типів обладнання виробничих систем і в системах холодного, гарячого водопостачання, в теплопунктах, котельних, системах пожежогасіння і системах каналізації.

Розробники апаратури автоматичного управління насосними групами все більше використовують засоби сучасної електроніки: мікропроцесорні блоки управління і електронні перетворювачі частоти (ПЧ). Використання обчислювальної техніки дедалі все більше переносить центр ваги зусиль розробників з апаратного на програмне забезпечення, а гнучкі можливості мікроконтролерів дають можливість застосовувати більш складні і досконалі алгоритми управління.

Ця стаття є викладенням досвіду розробок авторів по автоматичному управлінню насосними групами. Можливо вона допоможе програмістам, перед якими стоїть задача розробки програмного забезпечення індивідуальних проектів з насосним обладнанням, а також, безумовно, буде цікавою для наладчиків і користувачів шаф управління ЕЛЕКОН-11, які прагнуть глибокого розуміння роботи систем автоматичного управління.

В статті свідомо обійдені суто технічні питання програмної реалізації алгоритмів для конкретних зразків мікроконтролерів або промислових контролерів, щоб підкреслити повну інваріантність, незалежність алгоритмічної бази від конкретного засобу її реалізації.

Варіанти насосних груп розрізняються за застосуванням насосів, за кількістю насосів в групі і за способом електроживлення насосів.

За застосуванням насоси бувають:

1. Насосами холодного водопостачання – це напірні насоси, які повинні створити на виході певний тиск в системі холодного водопостачання (ХВП).
2. Підвищувальними – аналогічні насосам ХВП, але застосовуються в виробничих системах або для підвищення тиску у верхніх зонах гарячого водопостачання (ГВП) висотних будинків.
3. Підживлюючими – ці насоси є напірними з точки зору регулювання, але, як правило, працюють епізодично, щоб компенсувати втрати рідини (теплоносія) в системі.
4. Пожежними.
5. Циркуляційними – ці насоси створюють циркуляцію рідини-теплоносія (найчастіше вода або масло) в контурі нагрівання або охолодження. Циркуляційні насоси звичайно створюють невеликий перепад тиску (30÷100 кПа), в той час від напірних насосів звичайно потрібно створення тиску в тисячі і десятки тисяч кПа.
6. Дренажними.
7. Свердловинними.
8. Фекальними (в системах каналізації).

Для автоматизації дренажні, свердловинні і фекальні насоси алгоритмічно однакові. Вони повинні забезпечити відкачку рідини, але користувача не цікавить величина тиску на виході цих насосів.

Кількість насосів в групі звичайно від 1 до 6.

Метою об'єднання насосів в групи може бути:

- 1) Створення резерву.

- 2) Регулювання потужності насосної групи включенням певної кількості насосів.
- 3) Зменшення потужності (і вартості) перетворювача частоти, який застосовується для точного регулювання потужності насосної групи.

За способом електроживлення насоси бувають:

- 1) Насоси з живленням безпосередньо від промислової електромережі. Така схема живлення використовується коли непотрібно регулювання тиску в системі. Живлення насоса може включатися через магнітні пускачі або контактори або через електронні (найчастіше тиристорні) пускачі.
- 2) Насоси з живленням через перетворювачі частоти. Шляхом зміни частоти обертання регулюють тиск, перепад тиску, витрати рідини, що перекачується, або інші зв'язані з швидкістю обертання фізичні параметри.
- 3) Насосні групи з комбінованим електроживленням (частково через перетворювач частоти, частково – від електромережі).

Стан роботи насосів визначається за даними множини його датчиків, якими можуть бути:

- 1) датчик вихідного тиску насосної групи;
- 2) датчик наявності вхідного тиску, так званий датчик сухого ходу;
- 3) датчик достатнього вхідного тиску, коли підвищення тиску непотрібне і насоси можуть бути зупиненими;
- 4) датчики перепаду тиску на насосах (на виході насоса до його зворотного клапану – для насосів, які качають з резервуару);
- 5) датчики потоку або датчики витрат рідини;
- 6) контакти магнітних пускачів;
- 7) контакти теплових реле;
- 8) датчики перегріву обмоток електродвигуна;
- 9) датчики споживаного струму електродвигуна;
- 10) датчики вологості (для дренажних, свердловинних і фекальних насосів);
- 11) реле контролю послідовності і асиметрії фазних напруг;
- 12) сигнали спрацювання вбудованих систем захисту перетворювачів частоти.

Алгоритм управління насосною групою є послідовністю логічних і математичних дій обробки множини вхідних сигналів, в результаті якої визначаються:

- 1) дискретні (включити або відключити) сигнали управління електричними апаратами живлення – для насосів з живленням від електромережі;
- 2) аналогові (частота обертання) сигнали управління – для насосів з живленням від перетворювачів частоти;
- 3) аварійні стани насосу.

Наведений вище перелік застосування насосів, дані про їх кількість в насосних групах і можливі датчики сигналів показують необхідність множини алгоритмів управління. Алгоритм управління може бути розробленим відповідно до конкретної задачі застосування насосів, бути унікальним і повністю задовольняти потреби конкретного користувача. Проте накопичений досвід розробки алгоритмів і програм управління насосами дозволяє сформулювати загальну методику розробки алгоритму управління насосної групи. Перш за все, розглянемо класифікацію алгоритмів управління насосних груп, показану на схемі мал. 1.

Загальний алгоритм роботи насосної групи обіймає алгоритми управління окремих насосів, які здійснюють захист електродвигуна насоса (за даними датчиків його стану) і захист насоса в гідравлічній системі (за даними датчиків тиску, потоку і витрат рідини). До алгоритмів управління окремих насосів відносяться

також алгоритми їх ручного управління, яке здійснюється з пульта кнопочного управління або з пульта шафи автоматики насосів.

Від програм управління окремих насосів програма загального алгоритму роботи отримує інформацію про фактичний стан роботи кожного насосу в тому числі про вивід насоса з режиму автоматичного управління і наявність аварій насоса, яка визначає доступність насоса для автоматичного управління.

Потреба в роботі насосної групи визначається алгоритмом роботи об'єкта управління, фрагментом якого є насосна група, а також сигналами датчиків вхідного тиску (сухий хід або достатній тиск).



Мал.1. Класифікація алгоритмів управління насосних груп.

Кількість включених насосів і швидкість обертання насосів з живленням від ПЧ визначають продуктивність роботи насосної групи.

Визначення конкретних включених насосів виконується на підставі доступності насосів для автоматичного управління з урахуванням додаткових експлуатаційних вимог, найчастіше – вирівнювання використання моторесурсу насосів.

Ефективне вирішення інженерних задач вимагає комбінування всіх методів програмування [1]: об'єктноорієнтованого, автоматного [2], генетичного [3, 4], а часто стислі терміни виконання робіт потребують від виконавців наявності заздалегідь розробленої прозорої методики вирішення задачі. Наведена на мал.1 класифікація і є такою методичною заготовкою. Вона виникла в результаті накопичення досвіду розробки програм управління насосними станціями і групами для організації роботи програмістів, незалежно від засобів обчислювальної техніки, яка використовується для управління насосами в тому або іншому конкретному проекті автоматизації.

Розглянемо алгоритми відповідно до схеми мал.1.

1. Алгоритми захисту насосів.

Ці алгоритми відносяться до логічних алгоритмів. Їх реалізація потребує лише логічних операцій і витримок часу. Результатом захисту насосів є визначення відсутності або наявності аварій насоса (а не зупинки насосів – рішення про зупинку насоса приймається загальним алгоритмом роботи насосної групи. Наприклад, резервні насоси можуть працювати незважаючи на аварійні сигнали відсутності перепаду тиску на них, а якщо насоси пожежні, то ігнорувати навіть сигнали електричного захисту двигунів).

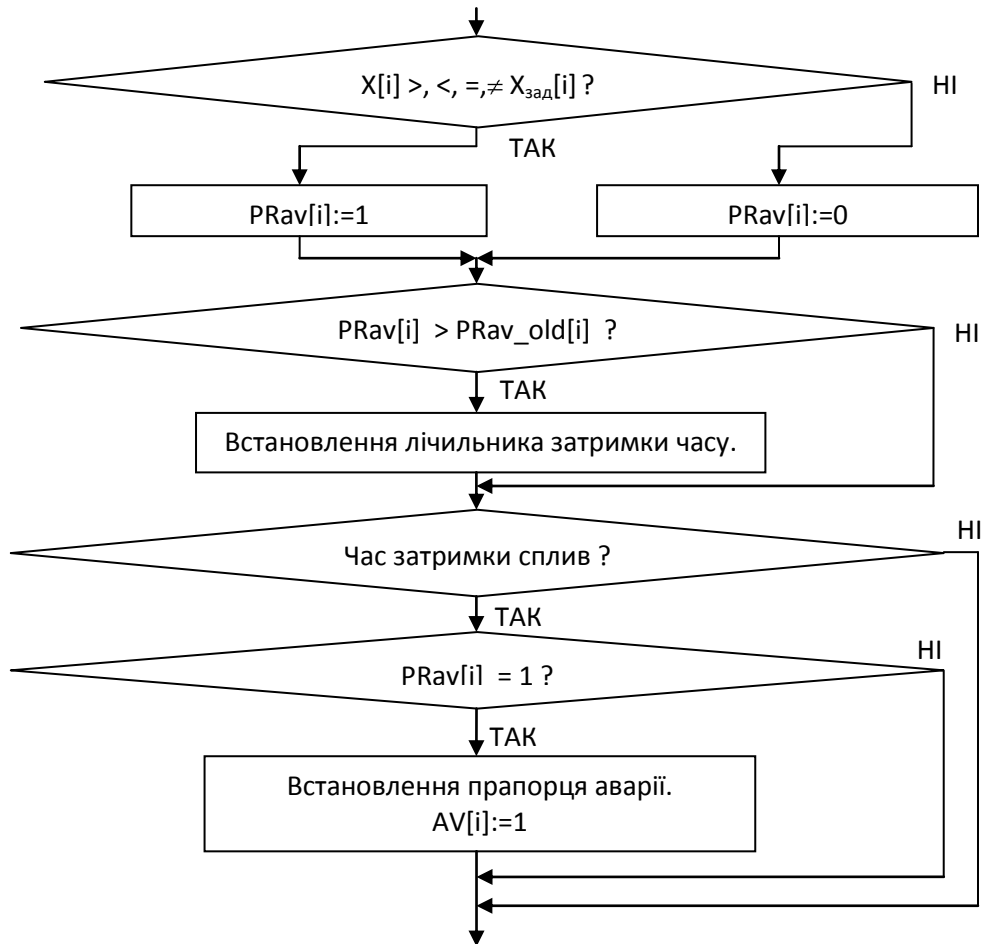
Аварії насоса визначаються складом контрольованих сигналів. З точки зору програміста визначення аварій – це визначення масиву прапорців, за значеннями яких насос буде визнаний непридатним або придатним до включення (роботи).

В більшості випадків це прості алгоритми за схемою мал.2.

На мал.2 індекс “i” підкреслює множину сигналів захисту і пов’язаних з ними прапорців (логічних змінних).

Алгоритм на мал.2 враховує можливі коливання аналогового або деренчання дискретного сигналу X за яким організується захист насоса.

Електричний або гідравлічний захист двигуна алгоритмічно однакові.



Мал.2. Типовий алгоритм захисту насоса.

Типовий алгоритм захисту насосів наведений на мал.2 – один з найпростіших фрагментів алгоритмічної бази управління насосами – наочно демонструє недолік графічного зображення схем алгоритмів. При ускладненні алгоритму схема швидко розростається і зображення алгоритму досягає не одного десятку сторінок, стає неоглядною, незручною для користування і виявляється фактором зменшення продуктивності праці при взаємодії алгоритміста (спеціаліст – технолог) і програміста (спеціаліст з обчислювальної техніки і програмування). Тому в практиці програмних розробок НТТОВ ЕЛЕКОН ми частіше використовуємо операторні записи алгоритмів (мал.4,5), подібні до записів C++, Paskal, Ada, то що.

2. Алгоритми ручного управління насосами.

Це логічні алгоритми. Цілком і однаково зрозумілі для програмістів і для користувачів. Щоб не переобтяжувати цю статтю очевидною логікою, залишимо їх без ретельного розгляду. Доречно зазначити лише наявність декількох,

найчастіше – двох режимів ручного управління в більшості розробок НТТОВ ЕЛЕКОН: 1) місцеве управління і 2) дистанційне управління.

Місьцеве управління – це управління з пульта кнопочного управління (ПКУ), який розташовується безпосередньо поряд з насосом в машинній залі або перемикач і кнопки ПКУ можуть бути змонтовані на шафі управління насосами, якщо вона розміщена в машинній залі в прямій видимості обладнання.

Дистанційне управління – це управління з пульта ШУ.

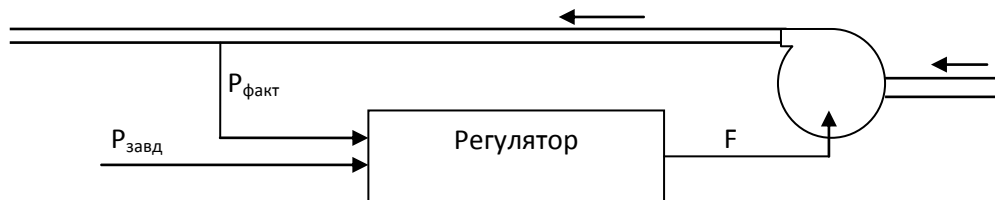
В системах диспетчерського управління може вводиться і третій режим – *ручне телеуправління* з пульта диспетчера.

Деякі режимів ручного управління дозволяють експлуатаційникам мати декілька пунктів здійснення ручного управління: 1) безпосередньо біля обладнання; 2) з електрощитової, де змонтовані шафи управління і 3) з диспетчерського пульта управління.

Цікавим є також питання ієрархії режимів управління. Найнижчий ступінь ієрархії має автоматичний режим управління, який в будь-який момент може бути відмінений. Наступним за ієрархією є місцевий режим управління – призначений для обслуговуючого персоналу об'єкта. Автоматичний або місцевий режим управління в будь-який момент можуть бути відмінені дистанційним режимом управління, призначеним для наладчика ШУ – особи яка несе відповідальність за всі дії, в тому числі і за безпеку роботи, і є пріоритетним суб'єктом управління. Вищий рівень ієрархії має режим блокування роботи насоса, коли перемикач його ПКУ встановлений в положення “-0-“, що дозволяє спостерігачу в машинній залі в будь-який момент блокувати дії оператора в дистанційному управлінні з електрощитової або ручне телеуправління диспетчера.

3. Визначення швидкості обертання насосів з живленням від перетворювачів частоти.

Виконується контуром системи автоматичного регулювання обертів насоса, мал.3.



Мал.3. Система автоматичного регулювання обертів насоса.

Регулятором може бути ПІД-регулятор або двохпозиційний регулятор.

ПІД-регулятор – це обчислювальний алгоритм визначення частоти обертання насосів за відомою формулою (див, наприклад, статтю “ПІД-регулятор” з Вікіпедії ru.wikipedia.org/wiki/ПІД-регулятор):

$$F_{упр} = Kn \cdot [K1 \cdot (P_{завд} - P) + \frac{1}{t_{инт}} \int_0^t (P_{завд} - P) \cdot dt + t_{диф} \cdot \frac{d(P_{завд} - P)}{dt}], \quad (1)$$

де $P_{завд}$ - завдання регулятора тиску, а P - фактично виміряний тиск;

Kn – коефіцієнт пропорційності (або підсилення);

$K1$ – штучно введений коефіцієнт для відключення пропорційної складової під час наладки ($K1=1$ – п-складова включена, $K1=0$ – п-складова відключена);

$t_{инт}$ – постійна часу інтегрування;

$t_{\text{диф}}$ – постійна часу диференціювання.

Для програмної реалізації, звичайно, потрібно виконати стандартну процедуру переходу від неперервних змінних $P=P(t)$ і t до дискретних – P_i і t_i та $\Pi_{\text{рег}}=t_i-t_{i-1}$ (період регулювання або час дискретизації). Для цього з (1):

$$F_{\text{упр}} = \int_0^t dF_{\text{упр}} = \int_0^t \left\{ K_{\Pi} \cdot K1 \cdot \frac{dP(t)}{dt} + \frac{1}{t_{\text{инт}}} \cdot [P_{\text{завд}} - P(t)] - t_{\text{диф}} \cdot \frac{d^2P(t)}{dt^2} \right\} \cdot dt$$

Якщо вважати $P_{\text{завд}}=const$, після підстановки $dP(t)=P_i-P_{i-1}$, $d^2P(t)=(P_i-P_{i-1})-(P_{i-1}-P_{i-2})$, $dt=\Pi_{\text{рег}}$, $dt^2=\Pi_{\text{рег}}^2$ і елементарних перетворень, знайдемо:

$$F_{\text{упр}} = \sum_0^i (\Pi + i + \Delta), \quad (2)$$

де $\Pi = K_{\Pi} \cdot K1 \cdot (P_i - P_{i-1})$ – додаток пропорційної складової;

$i = K_{\Pi} \cdot \frac{\Pi_{\text{рег}}}{t_{\text{инт}}} \cdot (P_{\text{завд}} - P_i)$ – додаток інтегральної складової;

$\Delta = K_{\Pi} \cdot \frac{t_{\text{инт}}}{\Pi_{\text{рег}}} \cdot (2 \cdot P_{i-1} - P_i - P_{i-2})$ – додаток диференційної складової.

Якщо вважати, що $P_{\text{завд}}=P_{\text{завд}}(t)$ змінюється, то позначивши відхилення фактичного значення від завдання $\varepsilon = P_{\text{завд}} - P$, одержимо:

$\Pi = K_{\Pi} \cdot K1 \cdot (\varepsilon_i - \varepsilon_{i-1})$;

$i = K_{\Pi} \cdot \frac{\Pi_{\text{рег}}}{t_{\text{инт}}} \cdot \varepsilon_i$;

$\Delta = K_{\Pi} \cdot \frac{t_{\text{инт}}}{\Pi_{\text{рег}}} \cdot (2 \cdot \varepsilon_{i-1} - \varepsilon_i - \varepsilon_{i-2})$.

В першому випадку, при розрахунку за величинами, матимемо асиметричну реакцію регулятора на збурення режиму – зміну P – і зміну завдання $P_{\text{завд}}$. В другому випадку, при розрахунку за відхиленням, реакція на збурення і зміну завдання – симетрична. У випадку асиметричної реакції, при зміні завдання регулятор буде поступово, з постійною часу $t_{\text{инт}}$, переходити з старого режиму роботи в новий, бо $P_{\text{завд}}$ враховується лише в інтегральній складовій. При симетричній реакції реагування за зміну завдання проходить точно так, як і при зміні регульованої величини. Перший варіант доцільно використовувати в системах, де завдання тиску є налагоджувальним параметром і змінюється вручну оператором стрибком. Другий варіант кращий, коли завдання тиску насосної групи автоматично визначається алгоритмом управління і плавно змінюється.

Робота ПІД-регулятора може бути перерваною *захистом від перевищення тиску* в системі. Для насосів холодного водопостачання при включенні станції на пустий вихідний трубопровід вихідний тиск тривалий час, поки вода витісняє повітря з системи, практично не росте, а коли повітря повністю витіснене водою, – різко зростає. ПІД-регулятор, який налагоджується на робочі коливання витрат води в системі, виявляється надто повільним, щоб швидко знизити оберти насосів. При перевищенні допустимого тиску частота обертання *регулюючого* насосу миттєво знижується до величини $F_{\text{упр}}=F_{\text{захисту}}$, а *допоміжні* насоси відключаються. Далі робота продовжується від мінімальної потужності.

Двохпозиційний регулятор – включає насос, коли $P \geq P_{\text{завд}} + \Delta P_{\text{гіст}}$ і відключає насос, коли $P < P_{\text{завд}} - \Delta P_{\text{гіст}}$, де $\Delta P_{\text{гіст}}$ – ширина зони гістерезису. Найчастіше двухпозиційний регулятор застосовується для підживлюючих насосів. Сигнал потреби включення насосів формується дискретним датчиком тиску (наприклад, ДЭМ-201). Для насосів ХВП житлових і адміністративних будинків такий регулятор може використовуватись, якщо система обладнана мембранним баком для економії електроенергії в години невеликих витрат води.

4. Визначення кількості включених насосів.

Визначення кількості включених насосів може бути дуже простим і прозорим в системі обладнаної датчиком витрат рідини, що перекачується. Якщо витрати рідини Q , то необхідна для задоволення потреб споживачів потужність $W=Q \cdot P_{завд}$. Якщо потужності насосів в насосній групі однакові і дорівнюють W_1 , то потрібна кількість включених допоміжних насосів $k_{доп}=Q \cdot P_{завд}/W_1$.

Проте витратоміри мають досить велику вартість і в більшості випадків застосування не встановлюються. Тоді і швидкість обертання регулюючого насоса і кількість включених допоміжних насосів визначається лише за даними датчика вихідного тиску. Ситуація подібна до регулювання режиму електричної мережі лише за даними вимірювань напруги.

Визначення кількості допоміжних насосів закладене також в програми серійних перетворювачів частоти, зокрема поширених FR-E740 відомої фірми Mitsubishi. В інструкціях ПЧ алгоритм визначення кількості допоміжних насосів названий "алгоритмом розширеного ПІД-регулювання". Далі будемо дотримуватись такої термінології. Нам достеменно невідомі деталі алгоритмів роботи розширеного ПІД-регулювання серійних перетворювачів частоти, не розкриті в інструкціях користувачів ПЧ. В наших розробках алгоритм розширеного ПІД-регулювання реалізований в двох варіантах.

Варіант 1. Якщо для регулювання одним насосом $F_{упр}$ змінюється в межах $0 \div F_{max}$ (де $F_{max}=50$ або 60 Гц залежно до стандарту електроживлення двигуна насоса), то для насосної станції в складі n однакових насосів $F_{упр}$ змінюється в межах $0 \div F_{max} \cdot n$. Кількість включених насосів k визначається як ціле результату ділення плюс одиниця $k=1+F_{упр}/F_{max}$. Частота обертання регулюючого насоса при цьому дорівнює залишку від ділення $F_{рег}= F_{упр} \% F_{max}$. Надто частих включень – відключень додаткових насосів на границях робочих діапазонів $F_{упр}= F_{max}, 2 \cdot F_{max}, 3 \cdot F_{max}, \dots$ запобігає видача сигналів включень насосів лише при перевищенні граничного значення на певну величину ΔF (звичайно $15 \div 16$ Гц).

Варіант 2. Полягає в визначенні окремого критерію включення і відключення чергового допоміжного насоса. При цьому частота обертів регулюючого насоса змінюється лише в межах частоти живлення одиночного регулюючого насоса $0 \div F_{max}$. Можливі два варіанти живлення багатонасосної станції: 1) коли кожний насос має свій власний ПЧ і 2) коли від ПЧ живиться лише один регулюючий насос, а допоміжні насоси включаються напряму від електромережі.

Якщо кожний насос живиться від окремого ПЧ, коли потужності регулюючого насоса недостатньо (частота його обертання досягла F_{max}) він перевизначається допоміжним, а наступний до включення допоміжний насос стає регулюючим і розрахунок ПІД-регулятора для нового регулюючого насоса починається з $F_{упр}=F_{min}$ (F_{min} може бути не рівним 0, а відповідати частоті, при якій насос працює на грані відкриття зворотного клапану при вихідному тиску $P=P_{завд}$). Коли потужність включених насосів надмірна, частота обертання регулюючого насоса знижується менше $F_{упр}<F_{min}$. Регулюючий насос перевизначається як допоміжний і виключається з роботи. Черговий до відключення допоміжний насос в цей час працює на максимальних обертах $F_{упр}=F_{max}$ і від цієї точки продовжується ПІД-регулювання зі зниженням частоти при зменшенні навантаження насосної станції.

Тобто у випадку живлення насосів "кожний від свого ПЧ" призначення насосів змінюються в процесі збільшення і зменшення навантаження насосної станції.

Цей алгоритм також придатний до випадку регулювання тиску лише кількістю включених насосів за призначеннями «робочій-допоміжні». Тут також, для зменшення переключень, призначення змінюються в процесі збільшення і зменшення навантаження насосної станції.

Коли від ПЧ живиться лише один регулюючий насос, а допоміжні насоси включаються напряму від електромережі і призначення насосів не можуть змінюватись, виконується маневр швидкістю регулюючого насоса: при включення допоміжного насоса – зменшення частоти до $F_{упр}=F_{min}$, а при відключенні – збільшення частоти до $F_{упр}=F_{max}$.

Як в першому, так і в другому випадку живлення допоміжних насосів, для надійного запобігання частих включень – відключень допоміжних насосів при граничних навантаженнях насосної станції, критерій включення – відключення допоміжного насоса визначається з двох складових:

- 1) досягненні граничної частоти $F_{упр}=F_{max}$ або $F_{упр}=F_{min}$;
- 2) включення допоміжного насоса затримується на заданий час, або, в більш гнучкому варіанті алгоритму, поки абсолютна величина значення інтегралу відхилення регульованого тиску не досягне встановленого значення:

$$abs(\int_0^t (P_{завд} - P) \cdot dt) \geq I_{зад} \quad (2)$$

Оскільки визначення конкретних включених насосів є задачею окремого алгоритму при переході значення $F_{упр}$ через величину $k \cdot F_{max}$ в першому варіанті і при виконанні критерію включення – відключення насоса в другому варіанті алгоритму розширеного ПІД-регулювання визначається лише кількість включених насосів в межах кількості доступних для автоматичного регулювання насосів (всі насоси станції, без насосів в ручних режимах управління і насосів в аварії).

5. Визначення конкретних включених насосів.

Потрібно зазначити, що викладення матеріалу п.4 для спрощення проведено за гіпотезою про те, що насосна група складається з однакових насосів. Таке дійсно має місце в переважній більшості проектів. Проте авторам практично доводилось розробляти ШУ і програми насосних станцій з насосами різної потужності. Відмінністю таких випадків є необхідність визначення постійних призначень насосів, які не можуть довільно змінюватись, тому деякі ускладнення визначення кількості потрібних допоміжних насосів в таких випадках відповідно ускладнювали і алгоритм визначення конкретних включених насосів. Але методика підходів до розробки алгоритмів управління не зазнавала корекції.

Для з'ясування питання про алгоритм визначення конкретних включених насосів розглянемо детальніше призначення насосів в насосній групі. Може бути три варіанти визначення призначень:

- 1) насоси можуть призначатись *робочими* і *резервними* ;
- 2) насоси можуть призначатись *регулюючими* і *допоміжними* – при регулюванні обертів за допомогою ПЧ або
- 3) *робочими* і *допоміжними* – при регулюванні потужності насосної групи лише кількістю включених насосів.

Робочим називається насос, який включається в роботу при потребі працювати насосній групі. Він працює до виникнення першої-ліпшої аварії.

Резервний насос залишається відключеним до моменту, поки робочий насос, внаслідок виникнення аварії, не буде зупиненим.

В насосній групі, яка складається з *робочих* і *резервних* насосів може бути декілька насосів одного призначення. При цьому всі *робочі* насоси включаються одночасно, а черговий *резервний* насос включається при черговій зупинці одного з *робочих* насосів.

Регулюючим називається насос призначений для підтримування вихідного тиску. Такий насос має засіб регулювання його потужності (частоти обертання) – в

більшості випадків ПЧ (хоча, теоретично можливе і застосування електродвигуна постійного струму – такі дорожчі, ніж асинхронний двигун з перетворювачем частоти).

Допоміжні насоси включаються до роботи при вичерпанні потужності *регулюючого* насоса, або відключаються, коли потужність насосної групи стає надмірною. Хоча додаткові насоси працюють з максимальною потужністю, живлення хоча б одного з них через додатковий перетворювач частоти дозволяє розробникам забезпечити повноцінне резервування при відмові перетворювача частоти регулюючого насоса.

Табл.1 показує використання варіантів визначення призначень для різного застосування насосів.

Табл.1. Використання варіантів призначень для різного застосування насосів.

Застосування насосів.	<i>Робочі – резервні.</i>	<i>Регулюючий – допоміжні.</i>	<i>Робочий – допоміжні.</i>
Насоси ХВП.	використовується	використовується	інколи
Підвищувальні.	використовується	використовується	інколи
Підживлюючі.	використовується	використовується	використовується
Пожежні.	використовується	не використ.	інколи
Циркуляційні.	використовується	використовується	інколи
Дренажні.	використовується	не використ.	використовується
Свердловинні.	використовується	не використ.	використовується
Фекальні	використовується	не використ.	використовується

Алгоритм управління насосів з призначеннями «*робочі – резервні*» відносно простий, виконувався ще за часів релейно-контактної техніки і при першому досвіді застосування мікроконтролерів просто копіювався розробниками.

В табл.2 показує логічні співвідношення сигналів при формуванні команди управління *робочого*, а табл.3 – *резервного* насосів.

Табл.2. Визначення команди включення робочого насоса.

Режим роботи.	Дозвіл роботи.	Спрацювання захисту.		Команда включення.	
		електричного	За тиском		
Автоматичний.	Є	нема	нема	Є	
			Є	нема	
		нема	-	-	нема
			-	-	нема
Блокування.	-	-	-	нема	
Ручний: місцевий або дистанційний.	Включений	нема	нема	Є	
			Є	нема	
		Відключений	-	-	нема
	-		-	нема	

Якщо в насосній групі більше одного *робочого* насоса, вони всі працюють за табл.2.

Якщо в насосній групі більше одного *резервного* насоса вони утворюють чергу з першого, другого і т.д. *резервних* насосів. Для першого *резервного* насоса «дозвіл роботи *робочого*» є дійсно дозволом роботи безпосередньо *робочого* насоса, а для другого *резервного* насоса визначається за відмовою першого *резервного* насоса або при відмові наступного *робочого*.

Табл.3. Визначення команди включення резервного насоса.

Режим роботи.	Дозвіл роботи.	Дозвіл роботи робочого.	Спрацювання захисту.		Команда включення.
			електричного	За тиском	
Автоматичний.	Є	нема	нема	нема	Є
				Є	Є
			Є	нема	нема
				Є	нема
	нема	-	-	-	нема
Блокування.	-	-	-	-	нема
Ручний: місцевий або дистанційний.	Включе- ний	-	нема	нема	Є
				Є	Є
			Є	нема	нема
	Є			нема	
Відключе ний	-	-	-	нема	

При переході до призначень «регулюючі – допоміжні» визначення таблиці включення насосів стають надто громіздкими, втім і при збільшенні кількості насосів в групі «робочі – резервні» за рахунок розростання логіки визначення сигналу «дозвіл роботи робочого».

Можна запропонувати простий методичний прийом заміни *призначень* на *пріоритети* насосів. Під *пріоритетом* насоса розуміється числове позначення *призначення* насоса. *Робочому* або *регулюючому* насосу призначається *пріоритет* 1, першому *резервному* або першому *допоміжному* – *пріоритет* 2, третьому – 3 і т.д. Якщо насос недоступний для автоматичного управління – *пріоритет* 0.

Таким чином *пріоритети* розширюють поняття *призначення* насоса визначенням можливості їх автоматичного управління.

Алгоритм мусить оперувати з двома масивами пріоритетів: *заданими* пріоритетами і *фактичними* пріоритетами.

Задані пріоритети визначають вихідний намір використання насосів з гіпотези про те, що всі насоси доступні для використання програмою управління. Така гіпотеза обмежує значення заданих пріоритетів величинами 1,2,3. Звичайно задані пріоритети підлягають гібернації* – запам'ятовуються енергонезалежно, без живлення, між відключенням і включенням ШУ.

Фактичні пріоритети визначаються в результаті корекції заданих пріоритетів залежно від стану органів управління насосів (і не підлягають гібернації). Для такого визначення відшукуються насоси в аварії, насоси в режимі блокування (ПКУ в -0-) і насоси зупинені в ручних режимах управління (ПКУ в -М-) і для них визначаються пріоритет 0 (*непрацездатний*). Після визначення нульових пріоритетів, значення фактичних ненульових пріоритетів змінюються. Фактичні пріоритети працездатних насосів перевизначаються натуральними числами в порядку зростання їх заданих пріоритетів. Наприклад: для насосної станції з 5 насосів задані пріоритети 3,2,1,4,5. Після визначення нульових пріоритетів у першого і третього насосів одержимо: 0,2,0,4,5. Якщо у кожного насоса свій

* - від лат. hibernation – “зимова сплячка”. Завдяки енергонезалежній пам’яті (RAM з батареєю, FLASH або EEPROM) відключення контролера стає подібним не до реінкарнації (смерть і нове народження) а до зимової сплячки тварин.

перетворювач частоти, після перевизначення пріоритетів: 0,1,0,2,3. Ускладнення алгоритму може бути, якщо не всі насоси живляться від перетворювачів частоти. Якщо, наприклад, від ПЧ живляться лише третій і четвертий насос, тоді, за непрацездатністю третього насоса, регулюючим може стати лише четвертий насос і від 0,2,0,4,5 перейдемо спочатку до 0,2,0,1,5 а потім до 0,2,0,1,3.

На мал.4 наведено типовий запис алгоритму визначення нульових пріоритетів групи з n насосів.

```
Цикл від i=0 до i=n-1, крок=1 {PRIORF[i]=PRIORZ[i];}
Цикл від i=0 до i=n-1, крок=1
{
    Якщо (АВАРІЯ_НАС[i] або ПКУ[i] в -0- або (ПКУ[i] в -M-)), то
        PRIORF[i]=0;
}
```

Мал.4. Визначення фактичних нульових пріоритетів насосів.

Масиви заданих і фактичних пріоритетів на мал.4 позначені відповідно $PRIORZ[i]$ та $PRIORF[i]$.

Дещо складнішим є алгоритм визначення фактичних пріоритетів. На мал.5 наведено такий фрагмент алгоритму трьох насосної станції з живленням кожного насоса від свого ПЧ.

```
k=0; Цикл від i=0 до i=2, крок 1 {Якщо PRIORF[i]==0, то k+=1;}
Якщо k==0, то // нема нульових пріоритетів
{
    Якщо EEpar_circl_estafet==0, то // якщо призначення змінюються по колу
    {
        // нема нульових пріоритетів
        Якщо (PRIORF[0]==1) та (PRIORF[1]==2) та (PRIORF[2]==3) ,то на TSTCHN;
        Якщо (PRIORF[0]==3) та (PRIORF[1]==1) та (PRIORF[2]==2) ,то на TSTCHN;
        Якщо (PRIORF[0]==2) та (PRIORF[1]==3) та (PRIORF[2]==1) ,то на TSTCHN;
        на NONULL;
    }
    Якщо EEpar_circl_estafet==1, то // якщо призначення змінюються по
    // естафеті
    {
        Якщо (PRIORF[0]==1) та (PRIORF[1]==2) та (PRIORF[2]==3) ,то на TSTCHN;
        Якщо (PRIORF[0]==2) та (PRIORF[1]==1) та (PRIORF[2]==3) ,то на TSTCHN;
        Якщо (PRIORF[0]==2) та (PRIORF[1]==3) та (PRIORF[2]==1) ,то на TSTCHN;
        Якщо (PRIORF[0]==1) та (PRIORF[1]==3) та (PRIORF[2]==2) ,то на TSTCHN;
        Якщо (PRIORF[0]==3) та (PRIORF[1]==1) та (PRIORF[2]==2) ,то на TSTCHN;
        Якщо (PRIORF[0]==3) та (PRIORF[1]==2) та (PRIORF[2]==1) ,то на TSTCHN;
        NONULL : PRIORF[0]=1; PRIORF[1]=2; PRIORF[2]=3; на TSTCHN;
    }
    Якщо EEpar_circl_estafet>1, то на TSTCHN; // якщо призначення змінюються
    // випадково
}
Якщо k==1, то // нульовий пріоритет у одного з насосів
{
    Цикл від i=0 до i=2, крок 1 {Якщо PRIORF[i]==0, то break;} // i=номер
    //насоса з нульовим пріоритетом
    Якщо i==0, то Якщо PRIORF[1]<PRIORF[2], то {PRIORF[1]=1; PRIORF[2]=2;
        на TSTCHN;}
        Інакше {PRIORF[1]=2; PRIORF[2]=1; на TSTCHN;}
    Якщо i==1, то Якщо PRIORF[0]<PRIORF[2], то {PRIORF[0]=1; PRIORF[2]=2;
        на TSTCHN;}
        Інакше {PRIORF[0]=2; PRIORF[2]=1; на TSTCHN;}
    Якщо i==2, то Якщо PRIORF[0]<PRIORF[1], то {PRIORF[0]=1; PRIORF[1]=2;
        на TSTCHN;}
        Інакше {PRIORF[0]=2; PRIORF[1]=1; на TSTCHN;}
}
```

```

Якщо k==2, то // нульовий пріоритет у двох насосів
{
    Якщо (PRIORF[0]==0) та (PRIORF[1]==0) ,то RIORF[2]=1;
    Якщо (PRIORF[1]==0) та (PRIORF[2]==0) ,то PRIORF[0]=1;
    Якщо (PRIORF[0]==0) та (PRIORF[2]==0) ,то PRIORZ[1]=1;
    На TSTCHN;
}
Якщо k==3, то на NORMAF;
TSTCHN : Цикл від i=0 до i=2, крок=1
{Якщо (PRIORF[i]!=PRIORFo[i]) та (PRIORFo[i]==0), то на LCHANGE;} //є зміна
//фактичних пріоритетів
На NORMAF;
LCHANGE: Цикл від i=0 до i=2, крок=1 {Якщо PRIORF[i]==1, то break; }
// регулюючий насос залишається регулюючим
Якщо EЕpar_circl_estafet==0, то // якщо призначення змінюються по колу
{
    Якщо i==0, то {PRIORZ[1]=2; PRIORZ[2]=3; на WRPRIOR;}
    Якщо i==1, то {PRIORZ[0]=3; PRIORZ[2]=2; на WRPRIOR;}
    Якщо i==2, то {PRIORZ[0]=2; PRIORZ[1]=3; на WRPRIOR;}
} інакше Якщо EЕpar_circl_estafet==1, то // якщо призначення змінюються по
//естафеті
{
    Якщо i==0, то Якщо PRIORZ[1]==2, то на NORMAF;
    Інакше {PRIORZ[2]=2; на WRPRIOR;}
    Якщо i==1, то Якщо PRIORZ[2]==2, то на NORMAF;
    Інакше {PRIORZ[0]=2; на WRPRIOR;}
    Якщо i==2, то Якщо PRIORZ[0]==2, то на NORMAF;
    Інакше {PRIORZ[1]=2; на WRPRIOR;}
}
Інакше на NORMAF;// при випадкових пріоритетах задані пріоритети не
//змінюються
WRPRIOR: Запам'ятовування масиву PRIORZ[3] для його гібернації;
NORMAF: ;

```

Мал.5. Визначення фактичних пріоритетів .

Автори розуміють, що детально розбиратись в мал.5 захоче лише програміст, який повинен виконати програмну реалізацію. Наладчиків і користувачів звичайно не будуть цікавити «нюанси».

Мал.4 і мал.5 демонструють операторний запис, який дуже близький до записів операторів алголо-подібних мов програмування. Заміна англійських ключових слів на українські (або російські) знімає психологічний бар'єр фахівців-технологів, які не займаються програмуванням, але є головними співучасниками створення алгоритму управління. Одночасно, програміст може бути зведений до рівня технічного інженерного працівника, якому залишається лише чітко перекласти на ту чи іншу мову програмування розробку алгоритміста. В багатьох випадках це нагально необхідно, бо фахівець з мікропроцесорів і програмування не може в стислі терміни стати кваліфікованим гідравліком, сантехніком, технологом... Оформлення записів алгоритму, які є сполучною ланкою взаємодії автора алгоритму – технолога і виконавця програмної реалізації – програміста, покладається на програміста. Технолог одержує інтуїтивно зрозумілий засіб контролю над програмуванням.

Операторний запис алгоритмів суттєво компактніший графічного зображення, про що вже сказано вище.

Мал.5 потребує коментаря відносно змінної *EЕpar_circl_estafet*, яка визначає порядок зміни пріоритетів. Зміна пріоритетів потрібна не лише після встановлення нульових пріоритетів (або навпаки при зникненні нульових пріоритетів – відновленні працездатності насосів – виконується з метою зменшення маневрів по включенню/відключенню насосів), але ще й для забезпечення рівномірного

виробітку моторесурсу насосів. Без зміни пріоритетів, в розглянутій трьох насосній станції найбільше зноситься регулюючий насос (пріоритет=1), а другий допоміжний насос (пріоритет=3) буде включатись до роботи лише епізодично. Зміна заданих пріоритетів за тим чи іншим правилом вирівнює знос насосів.

Можливий вибір: 1) зміна призначень «по колу» - **123**→**312**→**231**→123; 2) зміна призначень «по естафеті» - **123**→**213**→**231**→**132**→**312**→**321**→123 (по естафеті передається призначення регулюючого); 3) «випадкова» зміна пріоритетів, коли на початку роботи нормується лише величина пріоритету в межах 1÷3 і нерівність пріоритетів насосів, а для зміни пріоритетів виконується операція зсуву масиву заданих пріоритетів. Пріоритети будуть змінюватись також внаслідок аварій і переходів в ручні режими регулювання. В цьому випадку доречно згадати класичне філософське визначення, що “випадковість є непізнаною закономірністю”. Пріоритети зсуваються «по колу» з послідовністю, що склалася “по життю”.

Зміна заданих пріоритетів може виконуватись: 1) після відпрацювання насосної групою встановленого часу; 2) в заданий часовий момент доби або 3) коли різниця лічильника моторесурсу досягла встановленого значення.

Надбудова алгоритму управління насосної групи.

Опис алгоритмів управління насосами буде неповним, якщо не згадати про алгоритми, які можуть бути надбудовані над розглянутими вище технологічними алгоритмами найнижчого рівня. До таких алгоритмів відноситься:

- 1) алгоритм автоматичного квітування аварій і
- 2) алгоритм загального управління об'єкта автоматизації, до якого як частка входить насосна група.

Автоматичне квітування аварій – відміна аварії, зафіксованої самим автоматом управління. Може бути ефективним засобом поновлення нормальної роботи системи, втраченої внаслідок артефакту, коли обладнання виявилось працездатним. Прикладом може бути збій в роботі електроніки внаслідок удару блискавки. При цьому ПЧ або контролер ШУ можуть зафіксувати множину тих чи інших аварій, які не є наслідком пошкодження насосу, а лише наслідком електромагнітних наводок на датчики сигналів. Вдала спроба квітування аварії через 10÷20 с після її виникнення поверне автоматику до повноцінної роботи.

Можна виділити два типа алгоритмів автоматичного квітування аварій:

- 1) алгоритми часових інтервалів, коли квітування аварії виконується після того як сплив час періоду квітування, і
- 2) алгоритми квітування з аналізом доцільності квітування. В цьому випадку додатково контролюється, що на момент квітування, сигнали захисту обладнання прийшли до норми.

Алгоритм управління об'єкта автоматизації може мати декілька рівнів, особливо якщо об'єкт контролюється системою диспетчеризації і ШУ насосів і ШУ іншого обладнання утворюють розподілену обчислювальну систему.

Для об'єднання в такі системи, програма (алгоритм) управління насосної групи повинна мати прапорець дозволу роботи, подібний до прапорця дозволу роботи окремого насоса. Алгоритм вищого рівня лише визначає дозвіл або заборону роботи насосної групи. При цьому насосні групи відображуються в масиві програмного забезпечення програмними об'єктами. Програмування методів насосних груп і інших груп обладнання об'єкта автоматизації може виконуватись як програмування дискретних автоматів або методами генетичного програмування залежно від технічної доцільності. Робота кожної ШУ в системі диспетчерського управління визначатиметься масивом прапорців станів обладнання. Для насосних груп це прапорці дозволу їх роботи, які приймають лише два логічних значення:

дозволу або заборони роботи. Для іншого обладнання, наприклад, регулюючих клапанів прапорець стану може мати декілька логічних значень: дозвіл (заборона) автоматичної роботи, повне закриття клапану і його повне відкриття.

ПІДСУМКИ.

1. Наведений огляд алгоритмів управління насосами орієнтований на використання сучасних засобів обчислювальної техніки (мікроконтролерів, програмованих логічних матриць та ЕОМ).
2. Стаття є спробою авторів закрити прогалину в технічній літературі, що до алгоритмів управління насосами і дати конкретну методичку підходів до розробки алгоритмів управління насосними станціями, зрозумілу не лише фахівцям, знайомим з теорією програмування, а і проектантам і користувачам насосних станцій. Вузька тематична спрямованість виправдана широким розповсюдженням насосного обладнання в народному господарстві.
3. Метод визначення пріоритетів насосів і запропонований запис алгоритмів управління спрямовані на підвищення продуктивності спільної роботи технологів-алгоритмістів і програмістів має на меті забезпечення стислих термінів розробки алгоритмів автоматизації складних насосних станцій за індивідуальними проектами.

Література.

1. Непейвода Н. Н. Стили и методы программирования. М.: Интернет-университет информационных технологий, 2005.
2. Поликарпова Н.И., Шатыло А.А. Автоматное программирование, Санкт_Петербург, 2008.
3. Koza J. Genetic programming. On the Programming of Computers by Means of Natural Selection. MA: The MIT Press, 1998.
4. Гладков Л. А., Курейчик В. В., Курейчик В. М. Генетические алгоритмы. М.: Физматлит. 2006.