

Emil MICHTA

Instytut Metrologii Elektrycznej

Uniwersytet Zielonogórski

INTELLIGENTNE SYSTEMY POMIAROWO – STERUJĄCE

Streszczenie. W artykule przedstawiono charakterystykę i model komunikacyjny współczesnych systemów pomiarowo – sterujących (SPS). Postawiono tezę, że jedną z podstawowych cech inteligentnego systemu pomiarowo – sterującego (ISPS) jest zdolność do realizacji przetwarzania rozproszonego na poziomie węzłów SPS. Przedstawiono wyniki analiz wpływu przetwarzania rozproszonego na dotrzymanie ograniczeń czasowych. Zaprezentowano ISPS z przetwarzaniem rozproszonym typu regułowego oraz wskazano na nowe możliwości, jakie z punktu widzenia ISPS oferują sieci bezprzewodowe dedykowane dla SPS.

INTELLIGENT MEASUREMENT CONTROL SYSTEMS

Summary. In the paper, characteristics and communication model of a current measuring – control systems (MCS) is presented. It is stated, that the basic feature of an intelligent MCS (IMCS) are abilities to distributed processing implementation on nodes level. Results of analysis of distributed processing influence on tasks real-time deadlines are presented. IMCS with rule based distributed processing are outlined. At the end the new possibilities offered by industrial wireless networks for IMCS are presented.

1. WPROWADZENIE

O większości produkowanych obecnie czujników, przetworników i elementów wykonawczych możemy powiedzieć, że są to urządzenia inteligentne. Ich inteligencja jest wynikiem realizacji pewnych procedur przez mikroprocesor, dzięki którym urządzenia te potrafią realizować funkcje autoadaptacji, autokalibracji, autodiagnostyki, samokonfigurowania itp. Jeżeli z inteligentnych urządzeń pomiarowo – sterujących, które stanowią węzły systemu pomiarowo – sterującego (SPS), zbudujemy SPS, to czy taki system będzie można nazwać inteligentnym? Wydaje się, że stosowanie inteligentnych węzłów jest jedynie warunkiem koniecznym, ale niewystarczającym do zbudowania inteligentnego SPS (ISPS). W artykule przedstawiono wybrane właściwości funkcjonalne węzłów SPS, które umożliwiają budowę ISPS z inteligentnych węzłów. Postawiono tezę, że jednym z podstawowych warunków budowy ISPS jest zdolność węzłów systemu do realizacji przetwarzania rozproszonego. Przetwarzanie rozproszone oznacza zdolność węzłów systemu do realizacji zadań związanych z funkcjonowaniem obiektu, w którym zostały one zainstalowane. Przedstawiono model komunikacyjny i przykład funkcjo-

nowania ISPS z rozproszonym przetwarzaniem typu regułowego. Zwrócono uwagę na korzyści wynikające z wprowadzania energooszczędnych przemysłowych sieci bezprzewodowych do budowy ISPS.

2. ELEMENTY SPS

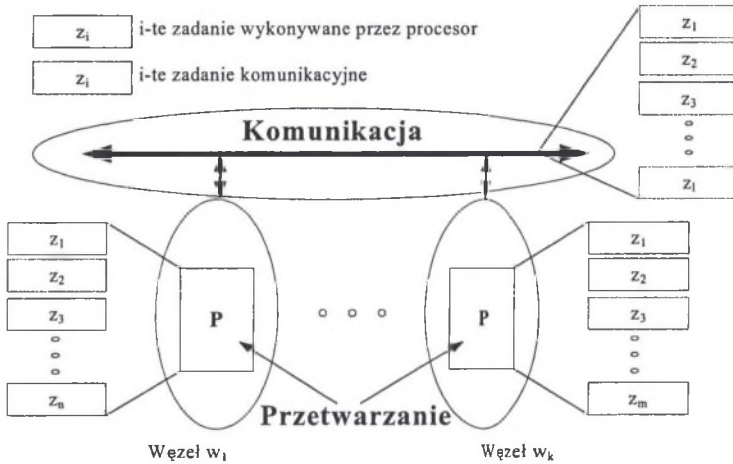
SPS jest zbudowany z węzłów w realizujących funkcje pomiarowe, sterujące i komunikacyjne w celu wykonania zadań postawionych systemowi. SPS jest określony przez węzły w_i oraz topologię fizyczną tf i topologię logiczną tl . Węzeł w_i określony jest poprzez zbiór niezależnych zadań Z realizowanych w węźle, metodę szeregowania zadań do wykonania (np. SC , RM , DM , EDF) oraz interfejs komunikacyjny. Zadanie z_i określone jest poprzez czas wykonania – C_i , okres występowania – T_i , ograniczenie czasowe – D_i .

Podstawowe topologie fizyczne spotykane w SPS to magistrała, gwiazda, gwiazda rozszerzona, pierścień i siatka. Dominującą obecnie topologią fizyczną SPS jest topologia magistrali bazująca głównie na standardzie RS-485. Obserwowany wzrost rozwiązań wykorzystujących Ethernet przemysłowy, Internet i transmisje bezprzewodowe np. w standardzie ZigBee powoduje, że wiele z powstających SPS na poziomie *Fieldbus*, *Devicebus* i *Sensorbus* wykorzystuje topologię gwiazdy lub gwiazdy rozszerzonej oraz siatki. Realizacja topologii siatki z wykorzystaniem nośników przewodowych jest zwykle trudna. W przypadku stosowania nośników bezprzewodowych realizacja tej topologii staje się znacznie prostsza. Z punktu widzenia ISPS stosowanie topologii siatki jest bardziej atrakcyjne, ponieważ pozwala ona w sposób naturalny na realizację takich funkcji systemu, jak przetwarzanie rozproszone, routowanie i tolerancja uszkodzeń. Podstawowe topologie logiczne spotykane w SPS to Master – Slave, Token Passing, CSMA/CA – wielodostęp z unikaniem kolizji, TDMA, Klient – Serwer, Producent – Konsument (PK) (*Publisher – Subscriber*). Z punktu widzenia potrzeb ISPS najbardziej interesujące są topologie CSMA/CA i PK. Dominująca obecnie topologia Master – Slave jest w wielu przypadkach modernizowana i po wprowadzeniu zmian może być również atrakcyjna dla rozwiązań ISPS. Przykładem takich modernizacji jest wprowadzenie w sieci Profibus-DP rozwiązania PK, pozwalającego na komunikację „Peer – to – Peer”.

3. MODEL KOMUNIKACYJNY SPS

Model komunikacyjny SPS przedstawiono na rys. 1. W modelu tym wyróżniono węzły pomiarowo – sterujące oraz interfejs komunikacyjny. Zarówno funkcje węzłów, jak i systemu komunikacyjnego podzielone są na zadania. W węzłach w_1, \dots, w_k mikroprocesory wykonują zadania z_i związane z obsługą czujników, elementów wykonawczych, przetwarzaniem danych pomiarowych oraz zadania komunikacyjne związane z obsługą wysyłanych i odbieranych danych do/z systemu komunikacyjnego. Przyjmuje się, że węzły są jednoprocessorowe i w danej chwili może być wykonywane co najwyżej jedno zadanie komunikacyjne. Zadania systemu komunikacyjnego realizują przenoszenie danych pomiędzy węzłami, a przenoszone dane umieszczane są w ramach komunikacyjnych. W systemie komunikacyjnym łączącym węzły w danej chwili może być realizowana jedna transmisja. Zadanie określone jest przez trójkę: $\{C, T, D\}$, gdzie C jest maksymalnym czasem wykonania zadania, T jest okresem jego występowania, a D jest ograniczeniem czasowym zadania. Parametry zadania T_i, C_i, D_i powinny spełniać warunek: $C_i \leq D_i \leq T_i$. Zarówno w węzłach, jak i w systemach komunikacyjnych

zadania do wykonania mogą być szeregowane w sposób statyczny lub dynamiczny. Wśród metod szeregowania statycznego najczęściej stosuje się metody szeregowania RM (ang. *Rate Monotonic*) uwzględniające okres występowania zadania, według zasady „krótszy okres – wyższy priorytet” oraz metody DM (ang. *Deadline Monotonic*) uwzględniające ograniczenie czasowe, według zasady „mniejsza wartość ograniczenia czasowego – większy priorytet”.



Rys. 1. Model komunikacyjny SPS

Fig. 1. MCS communication model

Na etapie wstępnym projektowania SPS istotne jest sprawdzenie warunku realizowalności zadań (1) przez procesor węzła i przez system komunikacyjny, ponieważ jego spełnienie jest warunkiem koniecznym dotrzymania ograniczeń czasowych stawianym zadaniom w projektowanym systemie [4,5].

$$\sum_{i=1}^N \frac{C_i}{T_i} \leq N \cdot \left(2^{\frac{1}{N}} - 1 \right) \tag{1}$$

Ponadto wykorzystując warunek realizowalności, można określić stopień wykorzystania procesora w danym węźle i obciążenie systemu komunikacyjnego projektowanego systemu. Czas odpowiedzi R zadania jest jednym z podstawowych parametrów w SPS interesujący zarówno projektantów SPS, jak i technologów. Określa on najgorszy przypadek czasu odpowiedzi, który jest liczony od chwili, kiedy zadanie jest gotowe do wykonania do czasu, kiedy zadanie zostanie wykonane. Maksymalną wartość czasu odpowiedzi R_{max} otrzymuje się wówczas, gdy jednocześnie aktywne będą wszystkie zadania o priorytecie wyższym niż priorytet danego zadania. Czas odpowiedzi R_i dla i -tego zadania wykonywanego w węźle przez procesor lub przesyłanego przez system komunikacyjny wyznacza się z zależności (2) [4].

$$R_i^{n+1} = \sum_{j \in hp(i)} \left(\left\lceil \frac{R_i^n}{T_j} \right\rceil \cdot C_j \right) + C_i \tag{2}$$

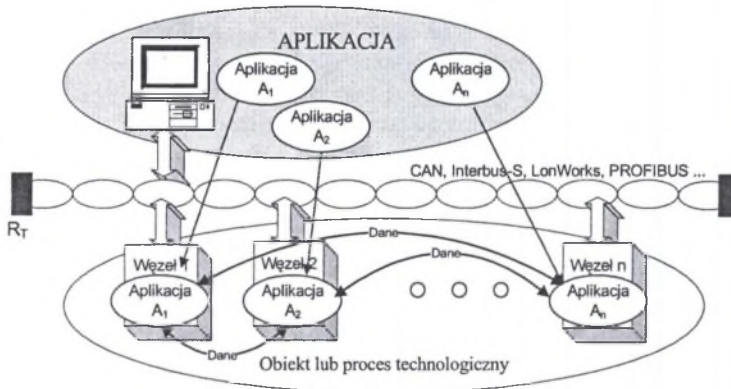
gdzie: $hp(i)$ oznacza zbiór zadań o priorytecie wyższym niż zadanie i .

Czas odpowiedzi zadania zawiera się pomiędzy $R_{i_{max}}$ i $R_{i_{min}}$ i powinien spełniać warunek:

$$R_{i_{max}} \leq D_i$$

4. PRZETWARZANIE ROZPROSZONE W SPS

Wprowadzenie inteligentnych węzłów oraz powszechne stosowanie komunikacji cyfrowej i protokołów komunikacyjnych realizujących model ISO-OSI do poziomu warstwy aplikacji umożliwiło realizowanie przetwarzania rozproszonego na poziomie węzłów sieci przemysłowej. Zadania realizowane dotychczas centralnie w stacjach nadrzędnych mogą zostać przeniesione do inteligentnych węzłów sieci przemysłowej. Zatem w węzle sieci przemysłowej, poza aplikacją opracowaną przez producenta urządzenia związaną z realizowanymi funkcjami pomiarowymi lub sterującymi, może zostać umieszczona aplikacja stanowiąca element przetwarzania rozproszonego. Aplikację tę przygotowuje użytkownik urządzenia i w fazie konfiguracji sieci przemysłowej przesyła ją do węzłów podłączonych do sieci. W przypadkach szczególnych może prowadzić to do sytuacji, kiedy warstwa wyższa nie jest niezbędna do realizacji aplikacji użytkownika. Jeżeli warstwy wyższe występują, to decentralizacja przetwarzania powoduje ich odciążenie, zmniejszając również obciążenie komunikacyjne sieci przemysłowej. Aplikacje rozproszone pracujące w poszczególnych węzłach wykorzystują informacje pomiarowo – sterujące z węzłów lokalnych oraz z innych węzłów pracujących w sieci przemysłowej. Na ich podstawie podejmowane są decyzje związane z węzłem lokalnym, a ponadto wynik przetworzenia aplikacji A_i pracującej w węzle w_i może być udostępniany innym węzłom (rys. 2) [2,3]



Rys. 2. Przetwarzanie rozproszone w SPS

Fig. 2. Distributed processing in MCS

Realizacja bardziej zaawansowanych funkcji na najniższym poziomie systemu informacyjnego z wykorzystaniem mocy obliczeniowej poszczególnych węzłów, bez potrzeby angażowania mocy obliczeniowych dostępnych na wyższych poziomach, wymaga zastosowania technik przetwarzania rozproszonego oraz określenia wpływu przetwarzania rozproszonego na obciążenie sieci przemysłowej i na dotrzymanie występujących ograniczeń czasowych. W tym celu wykorzystany zostanie opracowany wcześniej model komunikacyjny systemu i wyniki analiz systemu z wykorzystaniem elementów teorii szeregowania zadań.

Węzły sieci przemysłowej biorące udział w przetwarzaniu rozproszonym muszą być zdolne do wymiany danych pomiędzy sobą, bez udziału urządzeń pracujących na poziomach wyższych, co praktycznie ogranicza stosowanie przetwarzania rozproszonego do systemów z protokołami umożliwiającymi komunikację *peer-to-peer*, *token-passing* oraz *PK (Producent, Konsument)*.

Niezwykle istotnym zagadnieniem jest sposób definiowania zadania i jego dystrybucja do określonych węzłów sieci przemysłowej instalowanej na danym obiekcie lub procesie technologicznym. Zadania te najczęściej realizowane są przez użytkowników, zatem realizacja tych funkcji wymaga dysponowania odpowiednimi, prostymi w użyciu narzędziami programowymi do definiowania zadań dla poszczególnych węzłów, sprawdzenia poprawności zdefiniowanych zadań i ich dystrybucji do węzłów.

5. PRZETWARZANIE ROZPROSZONE TYPU REGULOWEGO

Węzły instalowane w SPS posiadają zdefiniowane przez producenta funkcje i parametry techniczne. W zależności od miejsca zainstalowania węzła różne jest zapotrzebowanie na sposób wykorzystania danych wytwarzanych lokalnie lub przez inne węzły pracujące w SPS. Zależy to od rodzaju realizowanej aplikacji przez użytkownika. W rozwiązaniach klasycznych aplikacje użytkownika były realizowane w warstwie nadrzędnej. W SPS z komunikacją przemysłowych *peer-to-peer* istnieje możliwość implementacji aplikacji użytkownika w warstwie pomiarowo – sterującej bez konieczności wykorzystywania warstw nadrzędnych. Jednym z prostszych sposobów przygotowania aplikacji jest wykorzystanie techniki przetwarzania regulowego, która jest bardzo zbliżona do typowego rozumowania człowieka [3]. Reguły składają się z dwóch części, warunkowej (*if*) oraz konkluzyjnej (*then*). Część warunkowa *if* określa, jaki warunek lub jakie warunki muszą być spełnione, ażeby mogły być podjęte czynności zawarte w części konkluzyjnej *then*. Reguły opracowywane są przez użytkownika przy użyciu programu narzędziowego wspomagającego ich tworzenie. Po utworzeniu zbioru reguł są one ładowane ze stacji nadrzędnej do węzłów pracujących w SPS. Po załadowaniu reguł do poszczególnych węzłów sieci są one natychmiast aktywne, a obecność stacji nadrzędnej pomocnej podczas tworzenia i dystrybucji reguł nie jest konieczna i może ona zostać odłączona od SPS.

Część warunkowa reguły może zawierać warunki odnoszące się do zmiennych sieciowych lokalnych, dostępnych w danym węźle oraz do zmiennych sieciowych, które znajdują się w innych węzłach. Część warunkowa reguły może się odnosić jedynie do zmiennej pomiarowej lub wejściowej zmiennej binarnej. Programy narzędziowe wspomagające tworzenie reguł powinny kontrolować poprawność semantyczną tworzonych reguł. W podobny sposób funkcjonuje część konkluzyjna, tzn. że konkluzje mogą się odnosić do elementów wykonawczych zainstalowanych lokalnie lub w innych węzłach SPS. Reguły załadowane do węzłów SPS są aktywne przez cały czas i w zależności od aktualnej wartości zmiennych sieciowych zdefiniowanych w polu warunkowym reguły sterują one polem konkluzji. Przetwarzanie danych zgodnie z określonymi regułami w poszczególnych węzłach nie jest synchronizowane.

Innym bardziej zaawansowanym sposobem przetwarzania rozproszonego, możliwym do realizacji w sieciach przemysłowych z wykorzystaniem techniki regulowej, jest przetwarzanie synchronizowane reguł, w którym reguły znajdujące się w węzłach SPS tworzą rozproszoną bazę wiedzy.

Do każdego z węzłów sieci przemysłowej może zostać załadowany różny zbiór reguł, który stanowi fragment bazy wiedzy obiektu lub procesu technologicznego. Ze względu na większą efektywność działania mechanizmu wnioskowania *do tyłu* forma zapisu reguł powinna mieć postać odwrotną do przedstawionej w edytorze reguł, tzn. reguła powinna się rozpoczynać od jej części konkluzyjnej, a kończyć częścią warunkową, np.:

Ri: a = "w_normie" if war₁ & war₂ & ... war_n.

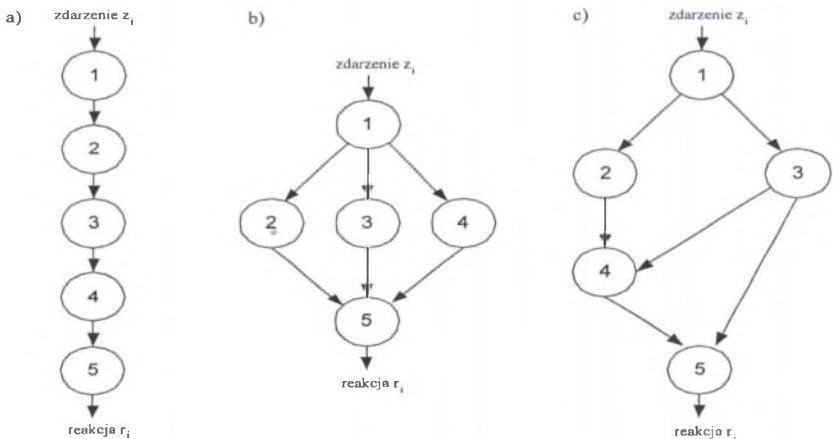
Przygotowanie reguł oraz ich dystrybucja odbywają się podobnie jak w przypadku poprzednio opisanym, natomiast w inny sposób realizowane jest przetwarzanie reguł. Po załadowaniu reguł do poszczególnych węzłów nie są one aktywne. Ich przetwarzanie jest synchronizowane wystąpieniem hipotezy $a = x?$ stawianej przez oprogramowanie użytkowe stacji nadrzędnej, przez użytkownika lub przez jeden z węzłów sieci przemysłowej. W przetwarzaniu biorą udział jedynie te reguły, których część konkluzyjna jest zgodna z postawioną hipotezą. W poprawnie skonstruowanej rozproszonej bazie wiedzy postawiona hipoteza powinna zostać potwierdzona co najwyżej przez jedną regułę. Tego typu przetwarzanie rozproszone można wykorzystać do realizacji bardziej zaawansowanych aplikacji, np.: wspomaganie prowadzenia obiektu lub procesu technologicznego, diagnostyka stanu obiektu lub jego podzespołów itp. W rozwiązaniach klasycznych tego typu aplikacje były realizowane na stacjach roboczych pracujących w warstwach nadrzędnych.

6. WPŁYW PRZETWARZANIA ROZPROSZONEGO NA DOTRZYMANIE OGRANICZEŃ CZASOWYCH

Przetwarzanie rozproszone w SPS oznacza wykonanie n zadań cząstkowych znajdujących się w różnych węzłach SPS powiązanych ze sobą w taki sposób, że tworzą one strukturę, która określa wzajemną współzależność pomiędzy zadaniami i kolejnością ich wykonywania. Wykonanie wszystkich zadań cząstkowych w określonej przez strukturę sekwencji kończy wykonanie przetwarzania rozproszonego [3]. Struktury utworzone przez sekwencję wykonywanych zadań mogą być bardzo różne: pionowe, poziome lub mieszane. Przykładowe struktury zadań realizowanych w ramach przetwarzania rozproszonego przedstawiono na rys. 3. Zadanie jest programem realizującym funkcję, wykonanie której jest niezbędne do wykonania całego przetwarzania rozproszonego.

6.1. Sekwencje przetwarzania rozproszonego

Przetwarzanie rozproszone inicjowane jest wystąpieniem zdarzenia z_i . Zdarzenie to inicjuje zadanie lub te zadania, które związane są z wystąpieniem zdarzenia z_i .



Rys. 3. Sekwencje przetwarzania rozproszonego
Fig. 3. Distributed processing sequence

Zakłada się, że w chwili wystąpienia zdarzenia z_i zadania cząstkowe przetwarzania rozproszonego inicjowane tym zdarzeniem znajdują się w węzłach. Kolejne zadania wykonywane są po zakończeniu wykonywania wszystkich zadań poprzedzających dane zadanie. Przetwarzanie rozproszone zainicjowane zdarzeniem z_i zostaje zakończone wykonaniem reakcji r_i . Wykonanie sekwencji zadań przetwarzania rozproszonego można zrealizować na wiele sposobów. Zadania te mogą być wykonane przez procesor jednego węzła lub mogą być wykonywane przez procesory znajdujące się w różnych węzłach. Znajomość czasu reakcji na występujące zdarzenia jest niezbędna w fazie projektowej do przeprowadzenia analiz czasowych projektowanego systemu pomiarowo – sterującego. Czas ten definiuje się jako czas liczony od chwili wystąpienia zdarzenia z_i do chwili wystąpienia reakcji r_i lub ciągu reakcji na to zdarzenie. W rzeczywistych układach czas reakcji R_{z_i} na zdarzenie z_i zmienia się pomiędzy:

$$R_{\min} \leq R_{z_i} \leq R_{\max} \quad (3)$$

Procesory pracujące w węzłach, w których realizowane są zadania przetwarzania rozproszonego, wykonują również inne zadania, a więc wykonanie zadania cząstkowego przetwarzania rozproszonego może być blokowane przez inne zadania. W rozwiązaniach praktycznych, do prowadzonych analiz np. związanych z dotrzymaniem warunków czasu rzeczywistego bardziej przydatna jest znajomość najbardziej niekorzystnego przypadku czasu reakcji na zdarzenie. Przyjmując, że poszczególne zadania przetwarzania rozproszonego realizowane są przez różne węzły i uwzględniając model komunikacyjny węzła przedstawiony w punkcie 2, czas reakcji na zdarzenie z_i wynosi:

$$C_{z_i} = \sum_{j=1}^n (t_{A_j} + t_{K_j}) \quad (4)$$

gdzie: t_{A_j} jest czasem aktywności zadania j w węźle,

t_{K_j} jest czasem przesłania wyniku przetworzenia zadania j do innych węzłów.

Czas aktywności t_{A_j} zadania w danym węźle liczony jest od chwili pojawienia się danych dla zadania w węźle do momentu rozpoczęcia transmisji wyniku przetworzenia. Jeżeli w węźle pracuje jeden procesor, to pojawienie się danych niezbędnych do rozpoczęcia wykonania zadania cząstkowego powoduje ustawienie tego zadania w kolejce wejściowej Q_{we} zadań do wykonania przez procesor P_j . Czas oczekiwania na rozpoczęcie realizacji zadania zależy od długości kolejki i sposobu obsługi zadań oczekujących w kolejce. Jeżeli wykonanie zadania rozproszonego będzie posiadało najwyższy priorytet i węzeł będzie pracował z wywłaszczaniem zadań, to czas oczekiwania w kolejce będzie krótki, ponieważ po pojawieniu się zdarzenia lub danych do wykonania procesor przerwie realizację aktualnie wykonywanego zadania i przejdzie do realizacji zadania przetwarzania rozproszonego. Natomiast w węzłach pracujących bez wywłaszczania zadań najdłuższy czas oczekiwania na wykonanie zadania rozproszonego będzie równy najdłuższemu czasowi wykonania wewnętrznego zadania realizowanego przez procesor.

Po wykonaniu zadania cząstkowego przetwarzania rozproszonego w węźle j wynik przetworzenia wysyłany jest do interfejsu komunikacyjnego. Czas oczekiwania wyniku przetwarzania rozproszonego w kolejce wyjściowej Q_{wy} zależy od liczby innych informacji wysyłanych z węzła j , od priorytetu nadanego wysyłanej informacji, od topologii logicznej interfejsu komunikacyjnego i od jego obciążenia przez pozostałe węzły SPS.

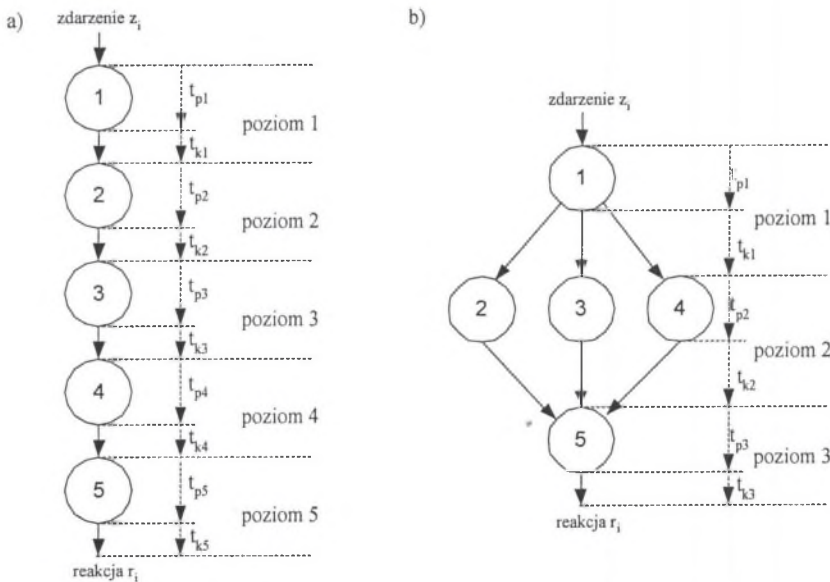
Rozważmy dla przykładu dwie struktury przetwarzania rozproszonego przedstawione na rys. 4. W obu przypadkach zadanie przetwarzania rozproszonego składa się z pięciu zadań cząstkowych, ale każde z nich tworzy odmienną strukturę, o różnej liczbie poziomów. Czas wykonania przetwarzania rozproszonego jest sumą czasów przeznaczonych na komunikację i

przetwarzanie. Zatem, całkowity czas potrzebny na wykonanie zadania rozproszonego można wyznaczyć z następującej zależności:

$$P = \sum_{i=1}^n t_{ki} + \sum_{j=1}^l t_{pj}, \quad (5)$$

gdzie: t_{pj} jest najdłuższym czasem wykonania zadania cząstkowego na poziomie j ,
 t_{ki} jest najdłuższym czasem przesłania wyniku zadania.

Struktura przetwarzania rozproszonego przedstawiona na rys. 4a jest strukturą pionową o pięciu poziomach. Na każdym poziomie przetwarzane jest jedno zadanie cząstkowe i realizowany jest jeden cykl komunikacyjny przesłania wyniku przetwarzania do kolejnego zadania. W strukturze przetwarzania rozproszonego pokazanego na rys. 4b wyróżniono 3 poziomy. Na poziomach pierwszym i trzecim przetwarzane jest jedno zadanie cząstkowe, natomiast na poziomie 2 wykonywane są równoległe trzy zadania cząstkowe. Na poziomie tym realizowane są trzy cykle komunikacyjne. W obu przypadkach wykonanie przetwarzania rozproszonego wymaga wykonania $n-1$ cykli komunikacyjnych. Różny jest natomiast czas poświęcony na przetwarzanie zadania rozproszonego. W pierwszym przypadku struktura przetwarzania rozproszonego jest sekwencyjna, zatem czas przetwarzania jest sumą czasów przetwarzania w poszczególnych węzłach. W przypadku drugim struktura przetwarzania jest pozioma, zatem występuje sytuacja, kiedy dwa lub więcej zadań cząstkowych przetwarzanych jest równoległe, dzięki czemu w strukturach poziomych uzyskuje się krótszy sumaryczny czas przetwarzania całego zadania. Jeżeli na danym poziomie struktury przetwarzania rozproszonego realizowane są dwa lub więcej zadań cząstkowych, to w prowadzonych analizach należy uwzględnić czas przetwarzania tego zadania, które wykonuje się najdłużej.



Rys. 4. Przykłady struktur przetwarzania rozproszonego
 Fig. 4. Examples of distributed processing

Przyjmując założenie, że czasy przetwarzania poszczególnych zadań cząstkowych i czasy komunikacji w obu strukturach (rys. 4a i 4b) są jednakowe, to całkowity czas przetwarzania rozproszonego dla tych struktur jest następujący:

$$P^{(a)} = \sum_{i=1}^n t_{ki} + \sum_{j=1}^l t_{pj} = \sum_{i=1}^4 t_{ki} + \sum_{j=1}^5 t_{pj} = 4t_k + 5t_p \quad (6)$$

$$P^{(b)} = \sum_{i=1}^n t_{ki} + \sum_{j=1}^l t_{pj} = \sum_{i=1}^4 t_{ki} + \sum_{j=1}^3 t_{pj} = 4t_k + 3t_p \quad (7)$$

Porównanie otrzymanych wyników przeprowadzono dla trzech następujących przypadków: $t_k \ll t_p$, $t_k \approx t_p$ i $t_k \gg t_p$.

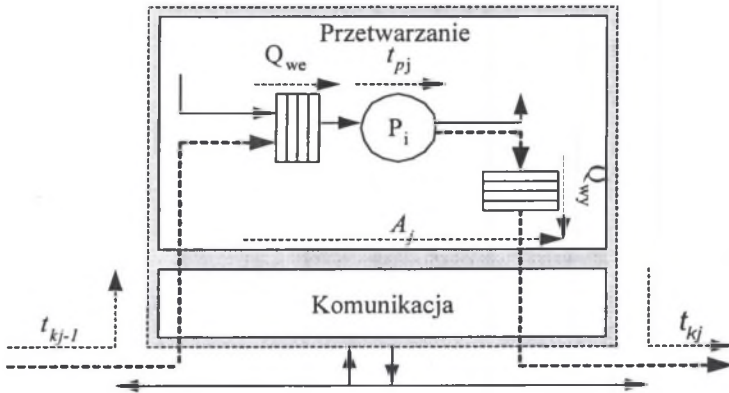
Dla $t_k \ll t_p$ czas realizacji przetwarzania rozproszonego zależy od czasów przetwarzania poszczególnych zadań cząstkowych i w strukturze sekwencyjnej (rys. 4a) wynosi on $P^{(a)} = 5t_p$ i jest 1,67 raza dłuższy niż w strukturze poziomej (rys. 4b).

Dla $t_k \approx t_p$ czas realizacji przetwarzania rozproszonego zależy w jednakowym stopniu od czasów przetwarzania i od czasów komunikacji. Czas przetwarzania rozproszonego w strukturze sekwencyjnej (rys. 4a) wynosi on $P^{(a)} = 9t_p$ i jest 1,28 raza dłuższy niż w strukturze poziomej (rys. 4b).

Dla $t_k \gg t_p$ czas realizacji przetwarzania rozproszonego zależy głównie od czasów komunikacji poszczególnych zadań cząstkowych i w obu rozpatrywanych strukturach wynosi on $P=4t_k$. Ocena rozpatrywanego przypadku prowadzi do następujących wniosków dotyczących przetwarzania rozproszonego w systemach o liniowej architekturze sieciowej:

- dla czasów komunikacji i czasów przetwarzania od $t_k \ll t_p$ do $t_k \approx t_p$ uzasadnione jest stosowanie przetwarzania rozproszonego,
- wzrost liczby zadań cząstkowych biorących udział w przetwarzaniu rozproszonym i rozbudowa struktury przetwarzania rozproszonego w kierunku poziomym uzasadnia stosowanie przetwarzania rozproszonego,
- jeżeli relacja pomiędzy czasem komunikacji i czasem przetwarzania dąży do warunku $t_k \gg t_p$, tzn. że czas wykonania zadania cząstkowego jest znacznie krótszy od czasu potrzebnego na przesłanie wyniku przetworzenia, to wówczas stosowanie przetwarzania rozproszonego nie jest uzasadnione.

Wykonane analizy prowadzono przy założeniu, że podczas realizacji zadania przetwarzania rozproszonego całe zasoby systemu przeznaczone były jedynie na realizację przetwarzania rozproszonego. W rzeczywistych warunkach taka sytuacja występuje bardzo rzadko, ponieważ systemy realizują wiele różnych zadań, które wykorzystują zasoby przetwarzające i komunikacyjne, a więc dostęp zasobów przetwarzających danego węzła i do zasobów komunikacyjnych sieci przemysłowej może być blokowany przez inne zadania, nawet w przypadku, kiedy zadania przetwarzania rozproszonego mają wysoki priorytet. Oznacza to, że w prowadzonych analizach poza czasem przetwarzania i czasem komunikacji należy uwzględnić inne czynniki mające wpływ na wyniki prowadzonych analiz. W tym celu posłużymy się rys. 5 przedstawiającym *i*-ty węzeł z procesorem P_i biorący udział w przetwarzaniu rozproszonym *j*-tego zadania cząstkowego, które jest aktywne w tym węźle przez czas A_j . Dane dla zadania cząstkowego wykonywanego w *i*-tym węźle po przesłaniu przez sieć trafiają do bufora sterownika komunikacyjnego, a następnie do procesora, który inicjuje zadanie cząstkowe przetwarzania rozproszonego. Zadanie to jest wykonywane bezpośrednio po odebraniu danych przez procesor z bufora sterownika komunikacyjnego lub trafia do kolejki Q_{we} zadań oczekujących na wykonanie.



Rys. 5. Przepływ danych w węźle z przetwarzaniem rozproszonym
 Fig. 5. Data flow in a node with distributed processing

W zależności od algorytmu pracy kolejki zadanie cząstkowe przetwarzania rozproszonego:

- ustawiane jest na końcu kolejki (algorytm FIFO),
- zajmuje miejsce w kolejce wynikające z przydzielonego mu priorytetu,
- oczekuje na zakończenie aktualnie wykonywanego zadania, jeżeli procesor wykonuje zadania bez ich wywłaszczania,
- jest wykonywane w pierwszej kolejności (pusty bufor FIFO lub najwyższy priorytet i procesor nie jest zajęty lub stosowany jest algorytm wywłaszczania).

Ponieważ praca węzła oraz wykonywanie zadania rozproszonego nie muszą być zsynchronizowane, więc w zależności od potrzeb dalsze analizy można prowadzić dla najgorszego przypadku, uwzględniając wartości średnie lub uwzględniając losowość zjawisk.

6.2. Wpływ przetwarzania rozproszonego na obciążenie sieci przemysłowej

W klasycznych systemach informatycznych wykorzystujących zasoby infrastruktury komunikacyjnej sieci komputerowych i telekomunikacyjnych oraz zasoby przetwarzające serwerów i stacji roboczych tworzy się modele bazujące na wartościach średnich lub modele probabilistyczne. Wartości średnie mogą zostać wyznaczone na podstawie obserwacji systemu lub na podstawie zakładanego jego średniego obciążenia. Stosując modele probabilistyczne, zakłada się, że czasy pojawiania się zadań do wykonania, odstępy pomiędzy nadejściem kolejnych zadań, czasy oczekiwania na wykonanie i czas oczekiwania na wysłanie wyniku przetwarzania są zmiennymi losowymi o znanych rozkładach prawdopodobieństwa, np. Poissona, Erlanga lub wykładniczym.

W odróżnieniu od klasycznych systemów informatycznych, które często opisuje się modelami otwartymi, sieciowe SPS wykorzystują dedykowaną strukturę komunikacyjną i są najczęściej systemami zamkniętymi o znanej liczbie węzłów i znanej liczbie zmiennych cyklicznych i sporadycznych. Wynika to ze specyfiki SPS, których funkcjonowanie bardzo ściśle związane jest z danym obiektem, procesem technologicznym lub aplikacjami informatycznymi użytkownika.

Prawidłowe funkcjonowanie obiektu, dla którego projektowany jest SPS, nakłada na projektowany system konieczność dotrzymania ograniczeń czasowych wynikających z cech obiektu, wymagań technologicznych i wykorzystywanych aplikacji informatycznych. Określenie warunków dotrzymania ograniczeń czasowych wymaga uwzględnienia w prowadzonych

analizach (w opracowanym modelu formalnym [4]) najgorszych przypadków. W odniesieniu do przetwarzania rozproszonego oznacza to wyznaczenie zależności na obliczenie najgorszego przypadku czasu przetwarzania rozproszonego. Czas przetwarzania rozproszonego mierzy się od chwili wystąpienia zdarzenia z_i inicjującego przetwarzanie rozproszone do czasu wykonania ostatniego zadania cząstkowego generującego reakcję r_i na zdarzenie z_i . Na czas przetwarzania rozproszonego może być nałożone ograniczenie czasowe D_i i wówczas poprawne funkcjonowanie systemu wymaga spełnienia warunku $P_r < D_i$. Zdarzenia z_i inicjujące przetwarzanie rozproszone mogą być generowane sporadycznie lub cyklicznie z okresem T_r . Często przyjmuje się, że ograniczenie czasowe jest równe okresowi występowania zdarzenia z_i .

Wykonanie przetwarzania rozproszonego o dowolnej strukturze wymaga wykonania $n-1$ cykli komunikacyjnych. Najgorszy przypadek czasu trwania i -tego cyklu komunikacyjnego w sieci przemysłowej pracującej bez wyłączenia wyznacza się z zależności rekurencyjnej [4] i ma następującą postać:

$$t_{ki}^{n+1} = B_i + C_i + \sum_{\forall k \in hp_i} \left\lceil \frac{t_{ki}^n}{T_k} \right\rceil C_k, \quad (8)$$

gdzie: C_i jest czasem wykonania i -tego zadania komunikacyjnego (czas transmisji i -tej ramki),
 T_k jest okresem występowania zadania k ,
 B_i jest czasem blokowania zadania i przez zadania o niższym priorytecie niż zadanie i ,
 hp_i jest zbiorem zadań o priorytecie wyższym niż dane zadanie i .

Czas blokowania B_i określa czas, przez jaki zadanie i o niższym priorytecie może opóźnić wykonanie zadania o priorytecie wyższym niż zadanie i .

Czas blokowania dla danego zadania określony jest w następujący sposób:

$$B_i = \max_{\forall k \in lp_i} \{ t_k \}, \quad (9)$$

gdzie: lp_i jest zbiorem zadań z priorytetem niższym niż zadanie i . W odniesieniu do cykli komunikacyjnych zadaniem jest przesłanie ramki.

Wykonanie przetwarzania rozproszonego składającego się z n zadań cząstkowych wymaga wykonania $n-1$ cykli komunikacyjnych, zatem najgorszy przypadek czasu trwania wszystkich cykli komunikacyjnych, zgodnie z podaną wcześniej zależnością wyniesie:

$$\sum_{i=1}^{n-1} t_{ki}^{n+1} = \sum_{i=1}^{n-1} (B_i + C_{r_i} + \sum_{\forall k \in hp_i} \left\lceil \frac{t_{ki}^n}{T_k} \right\rceil C_k). \quad (10)$$

Wyznaczenie najgorszego przypadku czasu trwania wszystkich cykli komunikacyjnych składających się na wykonanie przetwarzania rozproszonego wymaga znajomości czasu trwania, częstości występowania i priorytetu wszystkich cykli komunikacyjnych w analizowanym segmencie sieci przemysłowej. Jeżeli cykлом komunikacyjnym przetwarzania rozproszonego przydzielone zostały najwyższe priorytety, to najgorszy przypadek czasu trwania cykli komunikacyjnych wyniesie:

$$t_k = (n - 1) * B_{\max} + \sum_{i=1}^{n-1} C_{r_i}, \quad (11)$$

gdzie: B_{\max} jest czasem blokowania zadań komunikacyjnych przetwarzania rozproszonego przez zadanie o niższym priorytecie o najdłuższym czasie trwania.

Zależność powyższa jest prawdziwa przy założeniu, że częstość występowania zadania komunikacyjnego o najdłuższym czasie trwania B_{\max} jest nie mniejsza niż pojawianie się kolejnych zadań komunikacyjnych przetwarzania rozproszonego. W warunkach rzeczywistych czas t_k trwania cykli komunikacyjnych przetwarzania rozproszonego będzie zawierał się w następujących granicach:

$$\sum_{i=1}^{n-1} C_{r_i} \leq t_k \leq (n - 1) * B_{\max} + \sum_{i=1}^{n-1} C_{r_i}. \quad (12)$$

Lewa strona nierówności oznacza, że występujące podczas przetwarzania rozproszonego cykle komunikacyjne nie napotkają żadnych przeszkód podczas ich wykonywania. Jeżeli cyklem komunikacyjnym przetwarzania rozproszonego przydzielone zostały najniższe priorytety, to korzystając z wcześniej przedstawionych zależności najgorszy przypadek czasu trwania cykli komunikacyjnych przetwarzania rozproszonego wyniesie:

$$\sum_{i=1}^{n-1} t_{k_i}^{n+1} = \sum_{i=1}^{n-1} (C_{r_i} + \sum_{k=1}^m \left\lceil \frac{t_{k_i}^n}{T_k} \right\rceil C_k) = t_{k_{\max}}, \quad (13)$$

gdzie: m jest liczbą okresowych zadań komunikacyjnych w rozpatrywanym segmencie systemu.

W rzeczywistości czas wykonania zadań komunikacyjnych przetwarzania rozproszonego przy powyższych założeniach będzie zmieniał się w następujących granicach:

$$\sum_{i=1}^{n-1} C_{r_i} \leq t_{k_i} \leq t_{k_{\max}}. \quad (14)$$

Wykorzystując opracowane zależności, można dla danej struktury przetwarzania rozproszonego, dla danego węzła z określonym procesorem i czasem wykonania programu cząstkowego i sposobem szeregowania wszystkich zadań wykonywanych w węzle oraz dla danej sieci przemysłowej określić wpływ przetwarzania rozproszonego na dotrzymanie ograniczeń czasowych stawianych sieciowym systemom pomiarowo – sterującym na etapie ich projektowania.

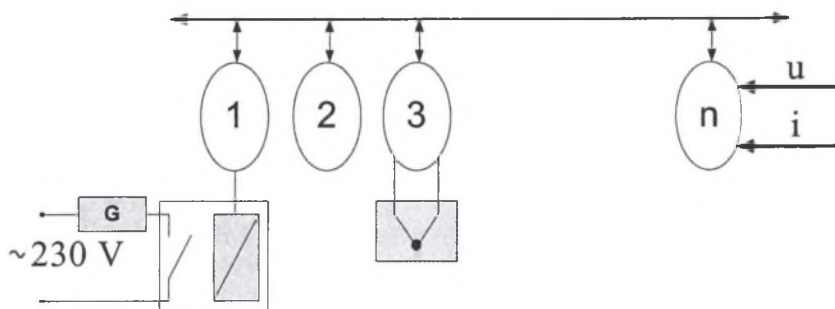
7. PRZETWARZANIE REGULOWE W ISPS

Podstawową kwestią związaną z implementacją przetwarzania rozproszonego na poziom węzłów ISPS jest sposób jego definiowania, dystrybucja zadań do węzłów oraz realizacja przetwarzania, w którym uczestniczą co najmniej dwa węzły. W jaki sposób definiować i realizować przetwarzanie rozproszone w SPS, ażeby było ono proste do wdrożenia dla użytkownika?

Do realizacji przetwarzania rozproszonego proponuje się wykorzystywać przetwarzanie typu regulowego z zastosowaniem wnioskowania "wstecz". Ten rodzaj wnioskowania nie angażuje zbyt mocno zasobów obliczeniowych. Opracowane reguły będą stanowiły formę reprezentacji bazy wiedzy o obiekcie umieszczonej w poszczególnych węzłach i niezbędną do realizacji między innymi zadań sterowania, diagnozowania, tolerancji uszkodzeń. Wiedzę o obiekcie

pozyskuje się od ekspertów, wykorzystując np. metodę arkuszy papierowych przygotowanych przez inżyniera wiedzy. Technikę przetwarzania regułowego zilustrowano na przykładzie SPS pokazanym na rys. 6. SPS zbudowany jest z n -węzłów połączonych siecią przemysłową. Węzeł w_1 jest węzłem wykonawczym i realizuje funkcje sterowania grzałką, natomiast węzły w_3 i w_n są węzłami pomiarowymi do pomiaru temperatury i mocy czynnej.

Przyjmijmy, że reguły przygotowane przez edytor reguł zostały podczas konfigurowania węzłów SPS umieszczone w poszczególnych węzłach. Przetwarzanie regułowe inicjowane jest zdarzeniem polegającym na postawieniu hipotezy przez jeden z węzłów SPS.



- Węzeł 1 - węzeł wykonawczy – sterowanie grzałką
- Węzeł 3 - węzeł pomiarowy – pomiar temperatury
- Węzeł n - węzeł pomiarowy – pomiar mocy

Rys. 6. Przykładowa struktura SPS
Fig. 6. Example structure of MCS

Przyjmijmy, że w rozważanym SPS węzeł wykonawczy w_1 stawia hipotezę „temp=?”, tzn. wysyła poprzez sieć przemysłową zapytanie do innych węzłów o aktualną wartość temperatury. W węźle 3 mierzącym temperaturę znajdują się trzy następujące reguły:

- R1: temp="w normie" if $t_{min} \leq temp \leq t_{max}$
- R2: temp="powyżej normy" if $temp > t_{max}$
- R3: temp="poniżej normy" if $temp < t_{min}$

Po odebraniu hipotezy węzeł 3 dokonuje przetworzenia reguł i wysyła odpowiedź, np.: temp="poniżej normy". Na podstawie uzyskanej informacji węzeł 1 przetwarza reguły umieszczone w tym węźle:

- R1: grzałka="załączona" if temp="poniżej normy"
- R2: grzałka="wyłączona" if temp="powyżej normy"

Jeżeli reguła R1 jest spełniona, to następuje załączenie grzałki. Pole warunkowe reguły może być bardziej rozbudowane i uwzględniać inne warunki, np. aktualny pobór mocy, co zilustrowano w poniższej regule:

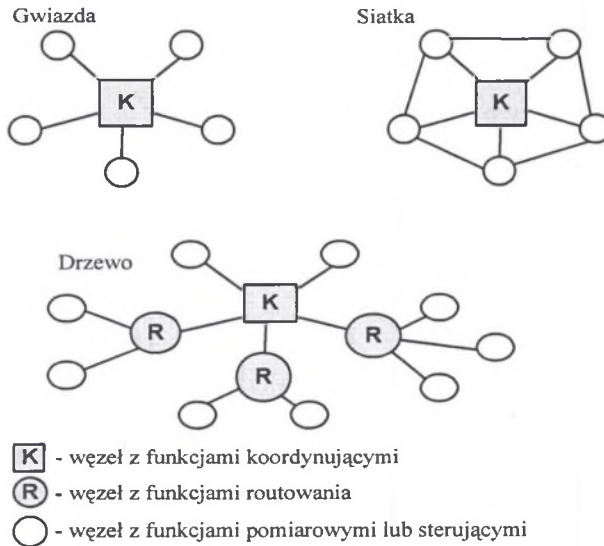
- R3: grzałka="załączona" if temp="poniżej normy" & (pobór mocy="w normie" or pobór mocy="poniżej normy")

Informacja o mocy czynnej pozyskiwana jest przez węzeł wykonawczy w podobny sposób jak pozyskiwanie informacji o temperaturze. W odróżnieniu od rozwiązań klasycznych SPS, w których surowe dane pomiarowe generowane są najczęściej w sposób cykliczny, w przypadku

przetwarzania regułowego przesyłana jest informacja o stanie wielkości mierzonej i jest ona przesyłana jedynie wówczas, kiedy jest ona niezbędna. Takie rozwiązanie odciąża system komunikacyjny od przesyłania dużej ilości surowych danych pomiarowych i pozwala w prosty sposób sparametryzować funkcje węzłów SPS.

8. ENERGOOSZCZĘDNE SIECI BEZPRZEWODOWE W ISPS

Z punktu widzenia zarówno projektanta, jak i użytkownika SPS stosowanie komunikacji bezprzewodowej jest interesujące ze względu na prostotę instalacji i rozbudowy oraz mobilność węzłów. W odróżnieniu od SPS z nośnikiem przewodowym sieci bezprzewodowe nie posiadają ograniczeń narzucanych przez fizyczną konfigurację systemu. Stosowanie transmisji bezprzewodowej pozwala na budowę sieci zarówno o topologii gwiazdy, jak i o topologii siatki lub topologii mieszanej (rys. 7). Podstawowe wymagania stawiane komunikacji bezprzewodowej we współcześnie opracowywanych SPS to: standaryzacja komunikacji, dostępność funkcji routowania, koordynacja węzłów, bezpieczeństwo, niski pobór mocy i integracja z Internetem.



Rys. 7. Topologie fizyczne w sieciach bezprzewodowych ZigBee
Fig. 7. Physical topologies in ZigBee Wireless Networks

W ISPS istotne są funkcje routowania i koordynacji oraz możliwość tworzenia sieci o topologiach mieszanych. Dostępność funkcji routowania w węzle oznacza, że węzeł może kierować przepływem informacji na podstawie adresu logicznego, dzięki czemu istnieje możliwość wyboru nowej trasy przesłania danych pomiarowych lub sterujących w przypadku wystąpienia uszkodzenia lub przeciążenia linii transmisyjnej. Nową ważną funkcją węzła w ISPS jest funkcja koordynacji, która pozwala na kontrolowane dołączanie/odłączanie węzłów pracujących w jednej domenie. Po dołączeniu nowego węzła lub usunięciu węzła uszkodzonego będącego w sieci, węzeł koordynujący informuje pozostałe węzły w domenie o zaistniałym zdarzeniu. Węzły z funkcjami koordynowania powinny realizować również funkcje routowania. Wymienione funkcje można zrealizować w sieci bezprzewodowej zgodnej z IEEE

802.15.4 i stosem protokołowym ZigBee [1,7]. Rozwiązanie to zostało zestandaryzowane w 2003 r. i charakteryzuje się ono bardzo niskim poborem mocy oraz bardzo prostą i tanią budową. Obsługuje urządzenia stacjonarne i ruchome. Parametry użytkowe tego standardu zależą od topologii (gwiazda, peer-to-peer, drzewo lub siatka) oraz pasma radiowego. ZigBee posługuje się czterema rodzajami ramek, tzn. koordynacyjną, danych, potwierdzenia oraz rozkazu. ZigBee stanowi bardzo dobrą alternatywę dla przewodowych systemów ograniczonych przez okablowanie. Pozwala tworzyć sieci o różnej topologii przede wszystkim dzięki zróżnicowaniu urządzeń na koordynujące, zwykłe oraz o ograniczonej funkcjonalności, którymi zazwyczaj są proste czujniki będące zarazem urządzeniami końcowymi.

W konkurencyjnej sieci bezprzewodowej Bluetooth możliwe są jedynie kilkumetrowe połączenia i praca w trybie *Master/Slave* z małą liczbą urządzeń. Bluetooth ustępuje ZigBee także pod względem szybkości tworzenia sieci *ad hoc* i rekonfiguracji istniejącej. Czas zarejestrowania nowego urządzenia w sieci ZigBee wynosi jedynie 30 ms. Obydwie technologie wykorzystują transmisję z rozproszeniem widma uzyskiwanym jednak różnymi metodami. W rozwiązaniu ZigBee zastosowano rozwiązania o niskim poborze mocy. Moc promieniowana jest poniżej 1 mW, dzięki czemu elementy sieci bezprzewodowej mogą korzystać z pary baterii nawet przez dwa lata. Wynika to także z charakteru transmisji, która jest dużo wolniejsza oraz nieciągła, tzn. poszczególne węzły sieci uzyskują dostęp do kanału i nadają tylko wtedy, kiedy wymaga tego aplikacja. Stos protokołowy ZigBee wynosi zaledwie 28 kB, zatem jego obsługę można zrealizować na prostych 8-bitowych mikrokomputerach jednoukładowych.

Według szacunków ceny układów do sieci bezprzewodowej ZigBee wynoszące obecnie ok. 5 USD spadną o połowę, co uczyni to rozwiązanie jeszcze bardziej cenowo atrakcyjnym. Przykładem komercyjnie dostępnej sieci bezprzewodowej kompatybilnej ze standardami IEEE 802.15.4 i Zigbee jest sieć bezprzewodowa SensorNet H900 firmy Sensicast [6]. Jest to rozwiązanie klasy „*end-to-end*” obejmujące zarówno sprzęt, oprogramowanie, jak i szeroką gamę interfejsów czujnikowych. Dwukierunkowa transmisja pomiędzy dowolnymi węzłami w sieci, dostępność funkcji routowania i koordynowania, możliwość budowania dowolnej topologii fizycznej oraz połączenia z sieciami przewodowymi Ethernet, LonWorks i BacNet czynią to rozwiązanie możliwe do wykorzystania w ISPS.

9. PODSUMOWANIE

Przedstawiony w artykule model komunikacyjny klasycznego SPS po modyfikacji polegającej na uwzględnieniu zadań wynikających z realizacji przetwarzania rozproszonego, koordynacji i routowania nadaje się do analizy ISPS. Zaprezentowany sposób realizacji przetwarzania rozproszonego typu regułowego potwierdza tezę, że możliwość realizacji przetwarzania rozproszonego jest warunkiem koniecznym do budowy ISPS. Pozwala ono na przesyłanie informacji, a nie nieprzetworzonych danych, poprzez co uzyskuje się zmniejszone wymagania na przepustowość systemu komunikacyjnego. Przedstawiona metodyka oceny wpływu przetwarzania rozproszonego na dotrzymanie ograniczeń czasowych pozwala na uzyskanie zarówno ocen ilościowych, jak i jakościowych dla poszczególnych zadań występujących w ISPS. Możliwości oferowane przez rozwiązania bezprzewodowe pozwolą na łatwiejsze wdrażanie takich cech ISPS, jak koordynacja węzłów i routowanie przesyłanych informacji.

Literatura

1. Alppi C., Ferrari S., Puri V., Sami M., Scotti F.: *New Trends in Intelligent System Design for Embedded and Measurement Applications*. IEEE Instrumentation and Measurement Magazine, June 1999, p. 36-44.
2. Huang G.T.: *Wireless sensors networks*. Technology Review, Cambridge Jul/Aug 2003, vol. 106, p. 50-55.
3. Kwiecień A.: *Analiza przepływu informacji w komputerowych sieciach przemysłowych*. Monografia. Studia Informatica, Vol. 23, Number 1 (47), Gliwice 2002.
4. Michta E.: *Przetwarzanie rozproszone w sieciach przemysłowych*. ZN Pol. Śl., seria Informatyka z.36, Gliwice 1999, s. 485-498.
5. Michta E.: *Modele komunikacyjne sieciowego systemu pomiarowo-sterującego*. Wyd. Pol. Zielonogórskiej, Zielona Góra 2000.
6. Werewka J., Żaba S.: *Projektowanie struktur sieci magistral miejscowych dla rozproszonych systemów sterowania*. Kwartalnik Elektroniki i Telekomunikacji, 1999, s. 25-50.
7. www.sensicast.com
8. www.zigbee.org

Recenzent: Prof. dr hab. inż. Jerzy Jakubiec

Wpłynęło do Redakcji dnia 7 grudnia 2004

Abstract

Characteristics and communication model of a current measuring – control systems (MCS) are presented. At the beginning MCS is defined. MCS is a set of nodes W which are running measurement, control and communication tasks. MCS is defined by nodes w_i and physical and logical topology. Node w_i is defined by set of independent task Z running in node. Node tasks are executed by processor according its priority. Tasks are scheduled by static or dynamic algorithms eg. *SC*, *RM*, *DM*, *EDF*. Task z_i is defined by executing time - C_i , period of activating - T_i , deadline - D_i . Communication model of the MCS is shown on fig.1. Equation (1) is used to verify tasks accomplishment for each node and for MCS communication system. Task response time is calculate by use of equation (2). It is stated, that the basic feature of an intelligent MCS (IMCS) are abilities to distributed processing implementation on nodes level. Results of analysis of distributed processing influence on tasks real-time deadlines are presented. Classical, hierarchical distributed processing architecture is presented on fig.2 and its symmetrical peer-to-peer architecture proposed for IMCS is shown on fig. 3. IMCS with rule based distributed processing are outlined. Simple example of IMCS (fig. 6) and set of rules for this system are discussed. At the end the new possibilities offered by industrial wireless networks for IMCS are presented.